

Entidades colaboradoras:

PROGESS, Proyectos y Gestión de Servicios Sociales
Universidad Estatal Península de Santa Elena

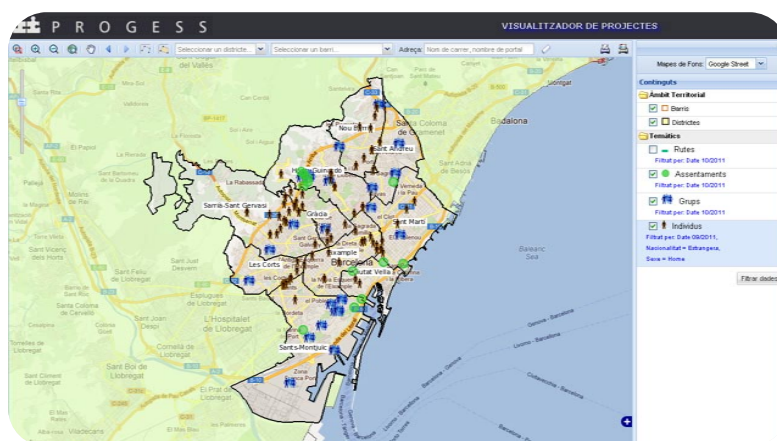
PROYECTO FINAL

“Sistema de Información Geográfico para visualización y consulta de datos del servicio de inserción social del ayuntamiento de Barcelona”

Autora: Karla Abad Sacoto

Tutores: Sonia Ambros Albesa (MTIG)

Robert Juan Francés (PROGESS)



Marzo 2012

Agradecimiento:

A Dios y a mis padres.

RESUMEN

En este documento se expone el trabajo realizado durante el desarrollo de un sistema de información geográfico como proyecto final del Máster en Tecnologías de la Información Geográfica. El sistema se realizó para y con la colaboración de la empresa PROGRESS y permitirá que los clientes de la empresa puedan, a través de un browser y conexión a internet, visualizar los resultados de los informes presentados por los educadores de calle. Además, pretende servir de referencia para otros proyectos similares y como herramienta de apoyo para futuros cambios o actualizaciones que deseen aplicar sobre el sistema realizado.

A través del documento se podrán evidenciar las fases de la elaboración del sistema desde su definición, declaración de objetivos y diseño, pasando por su implementación, la explotación y pruebas de los resultados. Finalmente se aportará con algunas conclusiones sobre el trabajo realizado.

Palabras Claves: sistema, geográfico, espacial, visualizador, fases.

ABSTRACT

This paper presents the development work done in a geographic information system as a final project of the Master of Geographic Information Technologies. The system is made to and with the collaboration of the company PROGRESS and enable to the enterprise's customers watch the results of the reports submitted by the street educators through a browser and Internet connection. It also aims to serve as reference for other similar projects and as a support tool for future changes or updates that wish to apply on the system performed.

Throughout the document will show the phases of system development from its definition, objectives statement and design, implementation, operation and testing results. Finally, some conclusions will contribute to the work done.

Keywords: system, geographic, spatial, geoPortal, phases.

INDICE

INTRODUCCIÓN.....	1
CAPÍTULO I	
1.1. ANTECEDENTES Y DEFINICIÓN:	2
1.2. OBJETIVOS	3
1.2.1. OBJETIVO GENERAL	3
1.2.2. OBJETIVOS ESPECÍFICOS	3
1.3. CONTENIDOS Y FUENTES DE INFORMACIÓN	4
1.4. REQUERIMIENTOS	5
1.4.1. FUNCIONALES	5
1.4.2. REQUERIMIENTOS NO FUNCIONALES	6
1.5. PLATAFORMA TECNOLÓGICA.....	6
1.6. ARQUITECTURA DEL SISTEMA.....	8
1.7. PLAN DE TRABAJO.	9
CAPÍTULO II	
2. DISEÑO.....	11
2.1. Diseño de la base de datos.....	11
2.1.1. Diseño Conceptual	11
2.1.2. Diseño Lógico	13
2.1.3. Diseño de datos espaciales	15
2.2. Diseño de la interfaz de usuario.....	17
CAPÍTULO III	
3. DESARROLLO E IMPLEMENTACIÓN	18
3.1. Implementación de la base de datos	18
3.2. Configuración y pruebas del servidor de mapas.....	20
3.2.1. configuración de MapServer.....	20
3.2.2. Elaboración de archivos del servidor de mapas.	21
3.2.3. Pruebas del servidor de mapas.....	23
3.3. Desarrollo e implementación de la aplicación cliente, visor de mapas.....	24
3.3.1. Descripción de la metodología de desarrollo utilizada.	24
3.3.2. Estructura y desarrollo de la aplicación.....	25
3.4. Implementación aplicación de actualización de datos.	27

CAPÍTULO IV

4. Elaboración y carga de datos.....	30
4.1. Transformación y carga de datos espaciales	30
4.2. Elaboración y Carga de datos alfanuméricos	32

CAPÍTULO V

5. Explotación y pruebas del sistema	33
5.1. Presentación de resultados.....	33
5.2. Pruebas del Sistema.	48

CONCLUSIONES.....	52
--------------------------	-----------

BIBLIOGRAFÍA.....	53
--------------------------	-----------

ANEXOS

ANEXO A: Guía de Instalación

ANEXO B: Manual de usuario aplicación de Mantenimiento

INDICE DE FIGURAS

Figura 1. 1 Arquitectura del Sistema.....	9
Figura 1. 2 Cronograma Plan de Trabajo	10
Figura 2. 1 Diagrama entidad – relación. Modelo Conceptual de la base de Datos.....	12
Figura 2. 2 Modelo Lógico de la base de Datos.	14
Figura 2. 3 Modelo de Diseño de GeoBase de datos ESRI®.	15
Figura 2. 4 Diseño interfaz usuario Visualizador web.....	17
Figura 3. 1 Creación BD Ingeniería Directa -Workbench.	19
Figura 3. 3 Verificación de extensiones - MapServer.....	20
Figura 3. 2 Tablas geometry_columns y spatial_ref_sys.	20
Figura 3. 4 Verificación conexión MapServer - MySQL.....	21
Figura 3. 5 Exportación a MapServer con Quantum GIS.	21
Figura 3. 6 Prueba servicio WMS servidor de mapas.	23
Figura 3. 7 Prueba servicio WFS servidor de mapas.	24
Figura 3. 8 Módulos Aplicación Web	27
Figura 3. 9 Ejecución comando de transformación de datos.	29
Figura 4. 1 PHPMyAdmin - comprobación de datos.....	30
Figura 4. 2 Comprobación de datos - Quantum Gis.....	31
Figura 4. 3 Comprobación de datos - MapServer.	31
Figura 5. 1 Mapa Base Distritos	33
Figura 5. 2 Mapa Base Google Satellite.	34
Figura 5. 3 Mapa Google Street.	34
Figura 5. 4 Simbolización de capas.	35

Figura 5. 5 Mapa a escala 1:54k.....	35
Figura 5. 6 Mapa a escala 1:27k.....	36
Figura 5. 7 Uso herramientas de navegación.	37
Figura 5. 8 Control árbol de capas.	37
Figura 5. 9 Selección de distrito.....	38
Figura 5. 10 Zoom a distrito seleccionado.....	38
Figura 5. 11 Selección de barrio.....	39
Figura 5. 12 Zoom a barrio seleccionado.....	39
Figura 5. 13 Herramienta localización por dirección.	39
Figura 5. 14 Resultado búsqueda por calle.....	40
Figura 5. 15 Filtro por individuos.....	41
Figura 5. 16 Resultado filtro por individuos.....	41
Figura 5. 17 Filtro por asentamientos.....	42
Figura 5. 18 Resultado filtro por asentamientos.	42
Figura 5. 19 Filtro por rutas.	43
Figura 5. 20 Resultado filtro por rutas.....	43
Figura 5. 21 Identificación elementos de asentamientos.....	44
Figura 5. 22 Identificación elementos de grupos.....	44
Figura 5. 23 Identificación elementos de individuos.	45
Figura 5. 24 Identificación elementos de rutas.....	45
Figura 5. 25 Herramientas de impresión mapa y leyenda.	46
Figura 5. 26 Impresión de mapa.	46
Figura 5. 27 Herramienta de impresión de leyenda.	47
Figura 5. 28 Impresión de leyenda.....	47
Figura 5. 29 Herramienta impresión de tabla.....	47
Figura 5. 30 Impresión tabla de elementos.	47
Figura 5. 31 Uso aplicación web con navegador Google Chrome.	48
Figura 5. 32 Uso aplicación web con navegador Firefox.....	48
Figura 5. 33 Uso aplicación web con navegador Internet explorer.	49
Figura 5. 34 Mensaje de error - campo dirección vacío.	49
Figura 5. 35 Mensaje de error - ingreso errado dirección.	49
Figura 5. 36 Mensaje de error - no encuentra dirección.	50
Figura 5. 37 Mensaje de error - errores en el formulario de filtro.	50
Figura 5. 38 Mensaje de error – no especifica fecha.....	50
Figura 5. 39 Mensaje de error - ingreso erróneo de fecha desde.	51
Figura 5. 40 Mensaje de error - ingreso erróneo de fecha hasta.	51

INDICE DE TABLAS

Tabla 1. 1 Fuentes de Información	4
Tabla 2. 1 Plantillas de diseño GeoBase de datos.....	16

INTRODUCCIÓN

Los sistemas de información geográfica representan una fuente importante e innovadora de información dado que en ellos se incluye la variable geográfica, permitiendo observar y hacer esquemas visuales de la ubicación de los datos y tomar mejores decisiones teniendo en cuenta estos parámetros. En los últimos años han tenido gran auge y muchas de las empresas han empezado a desarrollar aplicaciones que cuenten con este tipo de características o han buscado la manera de actualizar sus existentes sistemas para que las incluyan, esto debido principalmente a que han percibido la importancia y beneficio de contar con datos con representación espacial.

Por ello la empresa Progress – Proyectos y Gestión de Servicios Sociales ha solicitado la realización de una aplicación que permita visualizar la información de algunos de sus proyectos con localización geográfica. De esta manera sus clientes tendrán la posibilidad de revisar vía una aplicación web, disponible en todo momento y lugar, la información recogida por los educadores de calle de la empresa.

Así pues, este documento expone la realización del Sistema de Información Geográfico, elaborado para la empresa Progress y como herramienta de trabajo final del Máster en Tecnologías de la Información Geográfica del Departamento de Geografía de la Universidad Autónoma de Barcelona.

La realización del sistema fue dividida en varias fases las cuales se presentan a manera de capítulos en este documento.

En el primer capítulo se especifican la definición, objetivos y análisis de requerimientos del sistema. En el segundo capítulo se analiza y describe el diseño del sistema. Posteriormente, en el tercer capítulo, se reseña de manera global los criterios de implementación utilizados y se detalla unos cuantos parámetros considerados de mayor interés. En el cuarto capítulo se explican las metodologías utilizadas para la elaboración y carga de datos al sistema y el quinto y último capítulo exhibe los resultados y pruebas realizadas al sistema. Finalmente se presentan las conclusiones del trabajo realizado.

CAPÍTULO I

1.DEFINICIÓN Y ANÁLISIS DE REQUERIMIENTOS

1.1. ANTECEDENTES Y DEFINICIÓN:

El campo de los Sistemas de Información Geográfico ha evolucionado de manera ágil en los últimos años. Cada vez más se considera la importancia de contar con la ubicación espacial de la información que se produce además de manejar y en algunos casos difundir ésta información,

Según (Longley, Goodchild, Maguire, Rhind, 2011) los Sistemas de información Geográfica son una clase especial de sistemas de información que permiten realizar no solamente seguimiento de eventos, actividades y cosas sino también donde estos eventos, actividades y cosas ocurren o existen.

Coloquemos el ejemplo de una llamada de auxilio por incendio, si el departamento de bomberos contará con un sistema informático que les permita consultar la ubicación geográfica del incidente, calcular la ruta más corta de llegada y la ubicación de los suministros de agua las probabilidades de que su trabajo sea exitoso aumentaría en gran medida. De la misma manera existen diversas áreas en las cuáles contar con la ubicación espacial de la información puede llegar a ser incluso vital y muchas otras en las cuáles es de relevante importancia. Citando nuevamente a (Longley, Goodchild, Maguire, Rhind, 2011): **“Casi todo lo que sucede, sucede en algún lugar. Saber dónde pasa algo puede ser de importancia crítica.”**

Por ello la empresa Prograss – Proyectos y Gestión de Servicios Sociales ha solicitado la realización de una aplicación que permita visualizar y consultar la información de algunos de sus proyectos con localización geográfica. De esta manera sus clientes tendrán la posibilidad de revisar vía una aplicación web, disponible en todo momento y lugar, la información recogida por los educadores de calle de la empresa.

Políticas, planes de la organización sobre el manejo de la información:

Progress S.L. ha solicitado se aplique parte de la plataforma tecnológica que actualmente utilizan en otros proyectos informáticos. Particularmente, Apache como servidor Web, Mapserver como servidor de mapas y MySQL como gestor de bases de datos.

La información espacial y alfanumérica necesaria para el desarrollo y pruebas del proyecto ha sido entregada por la empresa.

Definición del proyecto de sistema de información a desarrollar

Luego de analizar los requerimientos de la empresa, la estructura actual de los datos y la información disponible se llegó a la conclusión de que lo más óptimo para obtener el producto que necesitaban era la realización de un sistema de información geográfico (SIG). Tras una reunión inicial de trabajo con el tutor de la empresa se llegó a definir el proyecto como un sistema de información geográfico que permite visualizar y consultar la información espacial y alfanumérica de algunos de los proyectos de la empresa, específicamente aquellos en los que trabajan los educadores de calle. A través de la primera reunión de trabajo con el cliente se ha dejado en claro el objetivo general y enfoque principal del trabajo.

1.2. OBJETIVOS

Después de definir a manera general el proyecto a realizar, se especifican los siguientes objetivos:

1.2.1. OBJETIVO GENERAL

El sistema de información tiene como finalidad principal ser instrumento de apoyo para la representación de resultados del trabajo de los educadores de calle, que se realiza en la empresa PROGRESS. En el ámbito territorial se limita al municipio de Barcelona.

1.2.2. OBJETIVOS ESPECÍFICOS

- Proveer la información básica territorial que permita situar y referenciar los objetos y hechos de interés del ámbito de servicios antes mencionado. En particular:
 - Divisiones administrativas

- Callejero
- Proveer la información temática territorial de interés. Definido por:
 - Individuos sin techo de la ciudad
 - Grupos de individuos sin techo de la ciudad
 - Asentamientos
 - Rutas empleadas por los educadores de calle
- Representar la dimensión espacial y la correcta simbolización de toda la información territorial.
- Soportar operaciones de de consulta y explotación de la información espacial y alfanumérica.
- Proveer de las herramientas de visualización necesarias para una satisfactoria utilización del sistema.
- Permitir generar fichas de resultados de las explotaciones realizadas.
- Proveer una interfaz de usuario que permita el autoaprendizaje del uso de la aplicación.
- Soportar mantenimiento y actualización de la información.

1.3. CONTENIDOS Y FUENTES DE INFORMACIÓN

Los datos necesarios para el cumplimiento de los objetivos planteados se lista y específica a continuación:

Grupo	Información	Cartográfica	Alfanumérica
Mapa Base	Mapa base de municipio de Barcelona	X	
	Mapa Satélite de Google	X	
	Mapa Calles de Google	X	
Divisiones Administrativas	Distritos	X	X
	Barrios	X	X
Callejero	Tramos de calles con asociaciones de rangos de dirección de portales.	X	X
Información temática	Individuos sin techo de la ciudad	X	X
	Grupos de individuos de la ciudad	X	X
	Asentamientos	X	X
	Rutas empleadas por los educadores de calle	X	X

Tabla 1. 1 Fuentes de Información

Esta información, a excepción de los mapas base, ha sido otorgada por PROGRESS para el desarrollo del proyecto y están en formato shape.

1.4. REQUERIMIENTOS

Los requerimientos son una descripción de las necesidades de un producto. Para el correcto funcionamiento del sistema es necesario se cumplan el siguiente conjunto de requerimientos:

1.4.1. FUNCIONALES

Los requerimientos funcionales establecen los comportamientos o funciones requeridas para el sistema, en este caso son:

- El sistema permitirá mostrar la información temática sobre tres tipos de mapas base: Google Satélite, Google callejero y distritos de Barcelona.
- Deberá mostrar una simbolización amigable de las diferentes capas de información territorial.
- El sistema podrá visualizar las distintas capas de información, en función de la escala.
- Proveerá de las herramientas básicas de visualización (zoom in, zoom out, pan, fullextent, escala), mapa guía y visualización de coordenadas x, y.
- Permitirá el control de la visualización de capas.
- En función del ámbito territorial al que pertenecen (barrios, distritos, calles), la aplicación deberá permitir la localización de la información temática.
- En función de los valores de sus atributos (por ejemplo, en caso de los individuos, según sexo o según origen), la aplicación deberá permitir filtrar la información temática.
- Se podrá identificar elementos de interés a través de una herramienta que mostrará los valores de los atributos relacionados.
- Se podrá generar una ficha en formato adobe reader con los resultados de las localizaciones y filtros.

1.4.2. REQUERIMIENTOS NO FUNCIONALES

Los requerimientos no funcionales se desarrollan sobre áreas claves de calidad, cada una de las cuales presentan características que permitirán ejecutar con eficacia y eficiencia las funcionalidades planteadas. Los requerimientos funcionales necesarios para el sistema son:

- El sistema deberá estar en capacidad de prestar servicio con niveles aceptables de desempeño y tiempos de respuesta.
- Debe garantizar soportar el crecimiento del volumen de la información almacenada en la base de datos.
- Se debe utilizar software de código libre para la realización del proyecto. En particular MapServer como servidor de mapas y MySQL como base de datos espacial.
- La interfaz de usuario debe ser lo suficientemente amigable para conseguir que el usuario final aprenda a utilizarla luego de 2 horas de entrenamiento.
- Ofrecer compatibilidad con los navegadores de uso más común.

1.5. PLATAFORMA TECNOLÓGICA.

Con el fin de cumplir con el requerimiento de utilización de software libre, se ha planteado la utilización de la siguiente plataforma tecnológica.

- **MapServer v5.6.6:** Servidor de Mapas.

Es una plataforma Open Source para la publicación de datos espaciales y aplicaciones de cartografía interactiva para la web. Originalmente desarrollado a mediados de la década de 1990 en la Universidad de Minnesota, MapServer es liberado bajo una licencia tipo MIT, y funciona en todas las principales plataformas (Windows, Linux, Mac OS X).

- MapServer es un proyecto de código abierto popular, cuyo propósito es la visualización dinámica mapas espaciales a través de Internet. Algunas de sus características principales incluyen:
- Soporte para la visualización y consulta de cientos de ráster, vector, y la base de datos de los formatos.

- Capacidad de funcionar con varios sistemas operativos (Windows, Linux, Mac OS X, etc).
- Soporte para lenguajes de scripting populares y entornos de desarrollo (PHP, Python, Perl, Ruby, Java, .NET).
- Aplicación de salida totalmente personalizable.
- Muchos productos listos para su uso abierto de entornos de aplicación.

En su forma más básica, MapServer es un programa CGI que se encuentra inactivo en el servidor Web. Cuando se envía una solicitud de MapServer, utiliza la información que pasa en la URL de la solicitud y el archivo de asignaciones para crear una imagen del mapa solicitado. La solicitud también puede devolver las imágenes de las leyendas, las barras de escala, mapas de referencia, y los valores pasados como variables CGI¹.

- **Gdal 1.8.0:** Librería que permite leer formatos geoespaciales (viene integrado con la versión utilizada de MapServer). Es una biblioteca de traductor para formatos de datos ráster geoespaciales que se distribuye bajo una licencia de código X / MIT estilo abierto por la Open Source Geospatial Foundation como una biblioteca, que presenta un único modelo de datos abstractos para la aplicación que llama a todos los formatos soportados. También viene con una variedad de utilidades de línea de comandos útiles para la traducción y procesamiento de datos².
- **Proj.4:** Librería que provee soporte de proyección a MapServer. Los usuarios que tienen sus datos en diferentes sistemas de proyección pueden utilizar esta librería para re proyectar en un sistema de proyección común. Ésta también es requerida para los servicios WMS, WFS, o WCS.
- **Fwtools 2.4.7:** Es un set de programas de código abierto para Sistemas de información Geográfica. Incluye herramientas para conversión de formatos de datos cartográficos.

1 MapServer Open Source Web Mapping (2011). «MapServer OverView». En: MapServer 6.0.1 Documentation. Disponible el 16-09-2011 en <http://mapserver.org/documentation.html>

2 Gdal GeoSpatial Data Abstraction Library (2011). «GDAL main page». En: GDAL Official Web Site. Disponible en <http://www.gdal.org/index.html>.

- **Quantun Gis 1.7:** Cliente Gis de escritorio que permite visualizar, manejar, editar y analizar datos de tipo raster y vectoriales, así como de bases de datos.
- **OpenLayers 2.10 y GeoExt:** Librerías script para el desarrollo de aplicaciones SIG Web.
- **Php 5.3.6:** Lenguaje de programación para aplicaciones web.
- **Ext:** Librería javascript para el desarrollo de aplicaciones web interactivas.
- **Apache 2.2.17:** Servidor de aplicaciones web.
- **MySQL 5.3.13 + spatial extensión:** Gestor de bases de datos alfanuméricas y espaciales. MySQL soporta las extensiones espaciales para permitir la generación, almacenamiento y análisis de elementos geográficos. Antes de MySQL 5.0.16, estas funciones estaban disponibles sólo para las tablas MyISAM. A partir de MySQL 5.0.16, InnoDB, NDB, BDB y file también son compatible con las funciones espaciales³.
- **PhpMyAdmin:** Aplicación web para gestionar las bases de datos de MySQL.

1.6. ARQUITECTURA DEL SISTEMA

En la figura 1.1 podemos observar la arquitectura propuesta para el sistema a desarrollar. Es una arquitectura de tres capas o niveles.

En la primera capa encontramos la base de datos espacial y alfanumérica que se encuentra implementada sobre el Sistema Gestor de Base de Datos (SGBD) MySQL el mismo que cuenta con una extensión para soportar datos espaciales.

³ MySQL Server Official Site (2011). «MySQL Spatial Extensions». En: MySQL Server Official Web Site. Disponible en <http://dev.mysql.com/doc/refman/5.5/en/spatial-extensions.html>.

En el segundo nivel se halla toda la lógica de negocio es decir las funciones que harán posible el tratamiento de los datos que se hallan en el primer nivel, aquí se encuentra el servidor de mapas MapServer que contiene todo el conjunto de funcionalidades para poder manejar información espacial: soporte para MySQL y servicios de producción de mapas como Web Map Service (WMS) y Web Feature Service (WFS) y además incluye un servidor Web Apache sobre el cual se desplegará la aplicación web cliente.

En la última capa se encuentran las interfaces de usuario y aplicaciones cliente que posibilitarán la interacción entre el usuario final y el sistema.

Estas capas interactúan entre sí, permitiendo la realización de las diferentes funcionalidades del sistema.

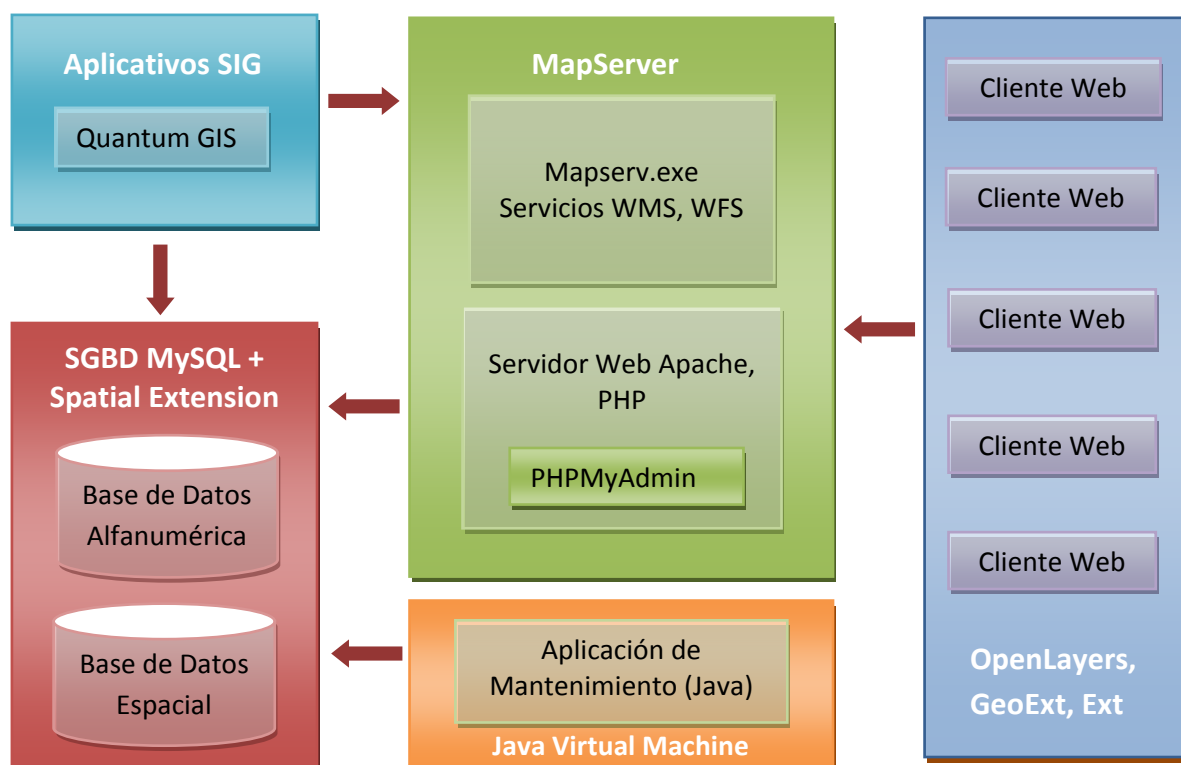


Figura 1. 1 Arquitectura del Sistema

1.7. PLAN DE TRABAJO.

Se estima que el proyecto constará de 17 semanas de desarrollo. El tiempo de desarrollo ha sido dividido entre las diferentes fases del proyecto. Algunas de las fases se realizan a la par y otras necesitan que la anterior sea terminada para poder iniciar su desarrollo. Tal como se puede observar en el diagrama de Gant mostrado a continuación.

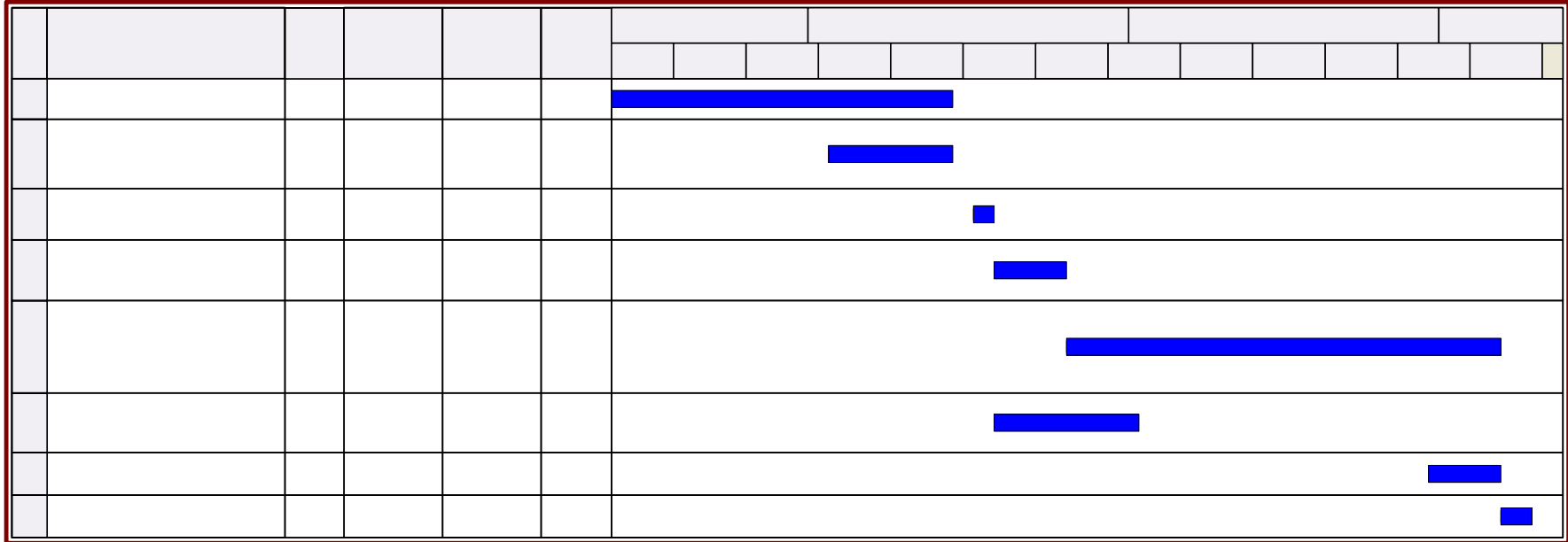


Figura 1. 2 Cronograma Plan de Trabajo

CAPÍTULO II

2. DISEÑO

2.1. DISEÑO DE LA BASE DE DATOS

Un modelo de datos es un conjunto de criterios construidos para describir y representar aspectos seleccionados del mundo real a través de un computador. Las decisiones acerca del tipo de modelo de datos a ser adoptado y del modelo en sí son vitales para el éxito de un proyecto SIG porque los tipos de análisis que pueden ser realizados son fuertemente influenciados por el modo en que el mundo real es representado.

Para representar el mundo real en computadoras es útil pensar en niveles de abstracción o generalización. Primero valorar la realidad como fenómeno del mundo real incluyendo todas las características que pueden o no ser percibidas por los individuos.

Segundo elaborar un modelo conceptual, el cual es un modelo orientado al humano que puede o no estar parcialmente estructurado y que además contiene objetos y procesos que se consideran relevantes para un problema en particular.

Tercero, se construye un modelo lógico, como representación de la realidad orientada a la implementación en ordenador, el cual es a menudo expresado a manera de diagramas. Y por último, se realiza el modelo físico que retrata la implementación de la base de datos y que frecuentemente comprende tablas almacenadas dentro de una base de datos.

En ésta sección se describirá el proceso de diseño de la base de datos del proyecto.

2.1.1. Diseño Conceptual

Para describir el diseño conceptual de la bases de datos se ha elaborado un diagrama entidad-relación que incluye las entidades participantes del modelo de negocio así como sus propiedades e

inter-relaciones. Este diagrama se construye en base al conocimiento que se ha adquirido luego de la fase de definición y es indiferente del Sistema Gestor de base de Datos (SGBD) a utilizar pues se realiza en un lenguaje enfocado al humano y no al ordenador. La siguiente figura ilustra el diagrama entidad-relación construido.

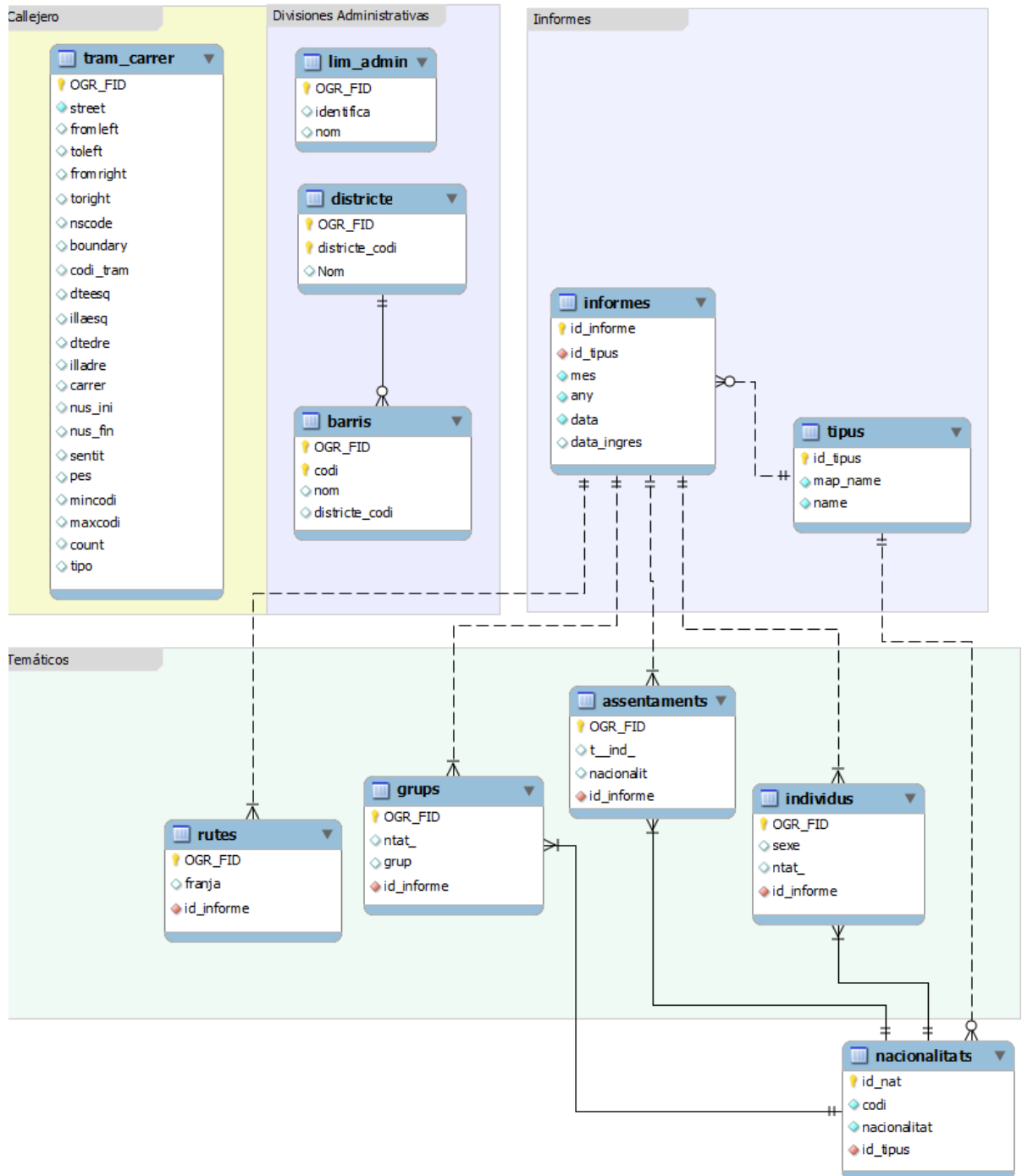


Figura 2. 1 Diagrama entidad – relación. Modelo Conceptual de la base de Datos.

El diagrama mostrado consta de entidades, relaciones y atributos.

Entidades: Se trata de cualquier objeto u elemento (real o abstracto) acerca del cual se pueda almacenar información en la base de datos. Las entidades presentes en el modelo conceptual propuesto se agrupan en cuatro categorías de acuerdo a la información que representan. La primera y segunda categoría contienen las entidades que permitirán ubicar la información sobre las divisiones administrativas y dirección postal dentro de Barcelona. En la tercera categoría se hallan las tablas que nos permitirán almacenar de manera ordenada los informes mensuales que ingresen los educadores de calle. Por último la categoría de temáticos contiene las entidades de los cuatro tipos de informes que el sistema será capaz de manipular, en éstas se encuentran los datos recolectados mensualmente por el personal de la empresa.

Relaciones: Representan asociaciones entre entidades. Las relaciones más relevantes en el diagrama son las que existen entre la entidad informe y cada una de las entidades del grupo temáticos. Éstas indican que cada informe agregado al sistema pertenecen a un solo tipo de información temática y que a su vez cada grupo de información temática está asociada a uno y solo un informe, el cual cuenta con fecha de ingreso, y mes - año al que pertenece los datos recolectados.

Atributos: Describen propiedades de las entidades y las relaciones.

2.1.2. Diseño Lógico

Este tipo de diseño representa los datos en un formato más cercano al lenguaje del ordenador. Esquematiza el modelo conceptual según el modelo concreto de un Sistema Gestor de Bases de Datos (por ejemplo Oracle o MySQL).

Como se describió en el capítulo anterior, en la sección de plataforma tecnológica, para el desarrollo del proyecto se utilizará el SGBD

MySQL por lo cual los tipos de datos utilizados en el diseño lógico son los soportados por este motor de base de datos. En la siguiente figura se esquematiza el diseño lógico de la base de datos.

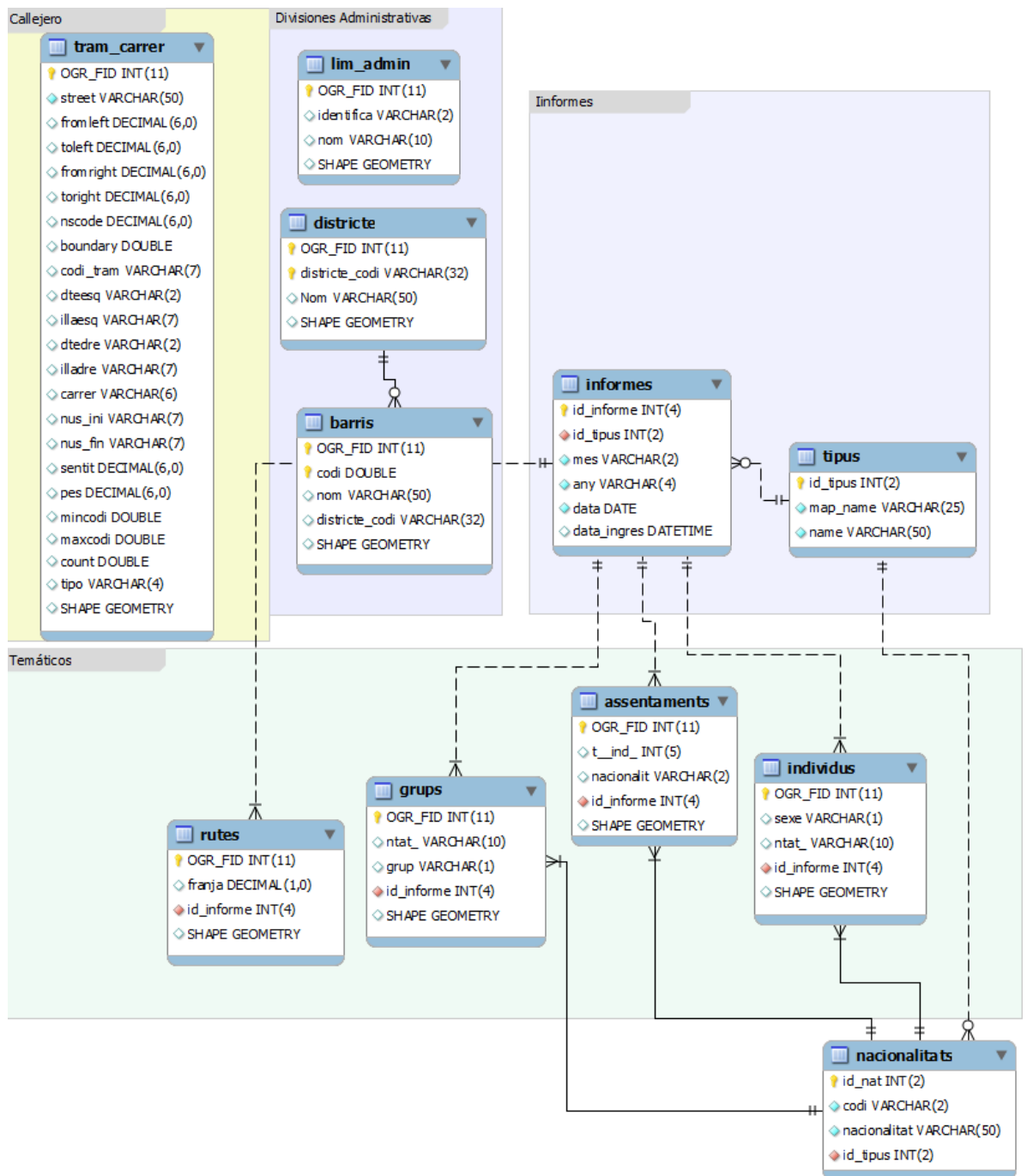


Figura 2. 2 Modelo Lógico de la base de Datos.

Es notable señalar que el diseño lógico parte del diseño conceptual revisado en la sección anterior. Ambos fueron realizados con la ayuda de la herramienta para manejo de bases de datos de MySQL, Workbench versión 5.2.34 CE.

2.1.3. Diseño de datos espaciales

Para el diseño de una base de datos espacial se han considerado las líneas guías para el diseño de GeoBase de Datos de ESRI® (Zeiler, 1999), el cual se describe a continuación.

Figura 2. 3 Modelo de Diseño de GeoBase de datos ESRI®.

La estructura de un conjunto de elementos, clases de elementos, grupos topológicos, las relaciones, y otros, permiten diseñar bases de datos geográficas que están cerca del modelo de lógico de datos.

Estos son los pasos básicos para diseñar una base de datos geográficos:

1. **Modelar vista de los datos del usuario.** Realizar entrevistas con los usuarios, entender la estructura de una organización, y analizar los requerimientos del negocio.
2. **Definir los objetos y relaciones:** Construir la lógica del modelo de datos con el conjunto de objetos y la forma en que están relacionados entre sí.
3. **Seleccionar la representación geográfica:** determinar si es vector, ráster, superficie o que representación de localización es mejor para los datos de interés.

4. **Adaptar a los elementos de la base de datos geográfica:** Ajustar los objetos en el modelo de datos lógicos en los elementos de una base de datos geográfica.
5. **Organizar la estructura de base de datos geográfica:** Construcción de la estructura de una base de datos geográfica con la consideración de grupos temáticos, asociaciones topológicas.

El diseño de la base de datos espacial se realizó a la par del diseño de la base de datos alfanumérica. Luego de tener el modelo conceptual de la base de datos (punto 1 y 2 del modelo de ESRI®) se analizó cuáles de las entidades participantes necesitarían representación espacial. Con la ayuda de las plantillas de diseño de GeoBase de datos (ver ejemplos en la tabla 2.1) se analizó el punto 3 del modelo de diseño nombrado y durante la realización del modelo lógico se ejecutó el punto 4 de lo cual tenemos como resultado el atributo SHAPE en las entidades de tipo espacial. Por otro lado dado que nuestra base de datos es de dimensión reducida no ha sido necesario ejecutar el punto 5 del modelo.

Entidad	Forma espacial	Tabla de atributos	Identificador	Atributos	Tablas relacionadas	Campos de relación
Divisiones Administrativas						
Limit	Polígono	Limit.shp	Identifica	OGR_FID, nom	lim_admin	
Districte	polígono	Districte.shp	districte-codi	OGR_FID, nom	districte	
Barri	Polígono	Barri.shp	Codi, districte-codi	OGR_FID, nom	barri	districte-codi

Entidad	Forma espacial	Tabla de atributos	Identificador	Atributos	Tablas relacionadas	Campos de relación
Temáticos						
Individus	Punto	individus.shp	OGR_FID	OGR_FID, sexe, ntat_, id_informe	informes	Id_informe
Grups	Punto	grups.shp	OGR_FID	OGR_FID, ntat_, grup, , id_informe	informes	Id_informe
Assentaments	Punto	assentaments.shp	OGR_FID	OGR_FID, t_ind_, nacionalit, id_informe	informes	Id_informe
Rutes	Punto	Rutes.shp	OGR_FID	OGR_FID, franja, id_informe	informes	Id_informe

Tabla 2. 1 Plantillas de diseño GeoBase de datos.

2.2. DISEÑO DE LA INTERFAZ DE USUARIO

En este apartado se describe, a través de la figura 2.4, el diseño preliminar de la interfaz de usuario del visualizador web (aplicación cliente) parte del sistema de información que se está desarrollando.

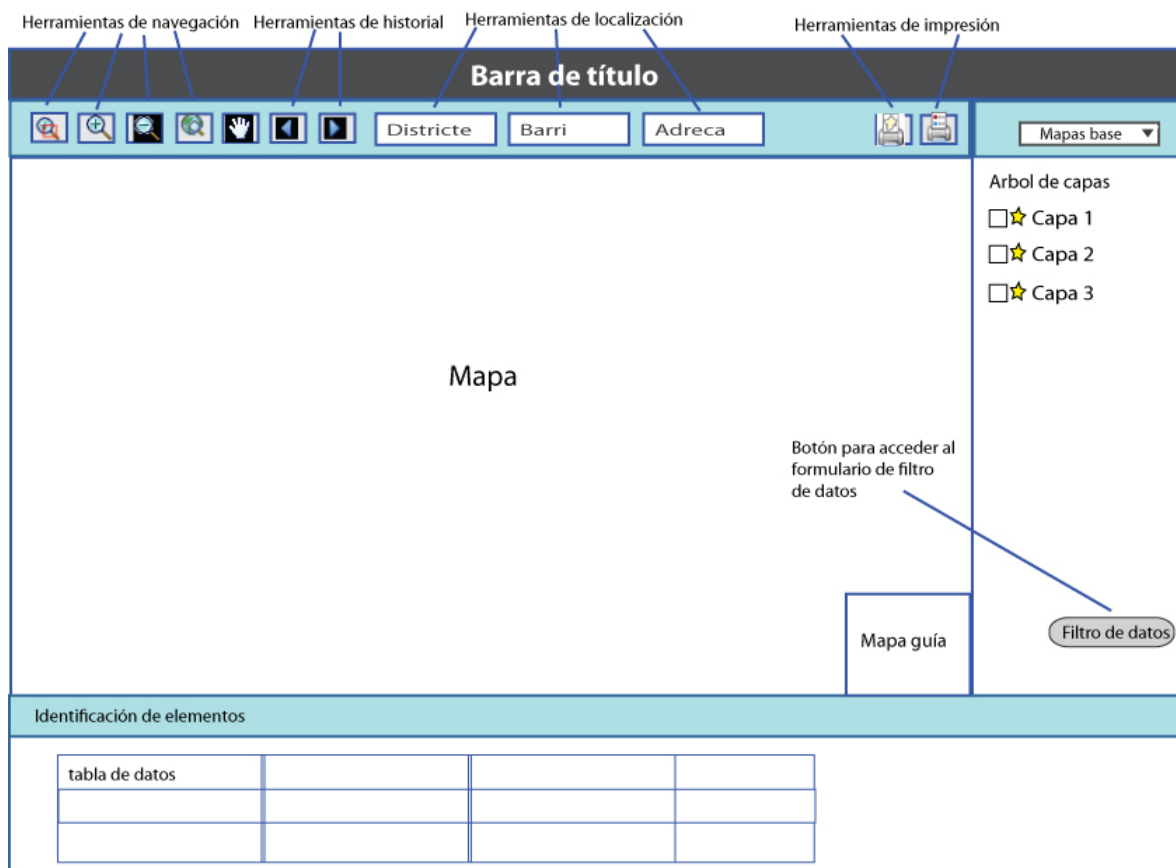


Figura 2. 4 Diseño interfaz usuario Visualizador web.

Como se ve en la imagen anterior, la interfaz de la aplicación contará con una parte central donde se visualizarán los mapas, en la parte superior se colocará la barra de título con el nombre de la aplicación y el logo y nombre de la empresa, además una barra de herramientas con las diferentes opciones de navegación, opciones de localización y de impresión. En el costado derecho de la interfaz se colocará el árbol de capas que se podrán visualizar sobre el mapa y bajo esta lista se hallará el botón que dará acceso al formulario para filtrado de datos. En la parte posterior de la interfaz colocaremos las tablas de elementos de las capas temáticas, las mismas que nos permitirán realizar la identificación de los elementos que se encuentran sobre el mapa.

CAPÍTULO III

3. DESARROLLO E IMPLEMENTACIÓN

3.1. IMPLEMENTACIÓN DE LA BASE DE DATOS

La programación de la base de datos es una de las partes más importantes dentro del desarrollo de un sistema informático debido a que es aquí donde se encuentra concentrada la lógica del negocio. Incluye la creación de la base de datos, las tablas participantes con sus diferentes atributos y tipos de datos, valores por defecto, relaciones y procedimientos de apoyo.

En este proyecto, la construcción de la base de datos se ha realizado sobre el motor MySQL versión 5.5.3 que cuenta con la extensión espacial la cual nos permite almacenar datos de tipo espacial dentro de la base.

Para implementarla se ha utilizado la herramienta de ingeniería directa (forward engineering) que provee Workbench, utilizando como base el diseño lógico de la base de datos.

La ingeniería directa reduce el tiempo de creación de una base de datos ya que será la herramienta de diseño quien genere los códigos DDL dentro del DBMS⁴.

La figura 3.1 muestra el proceso del uso de la herramienta para ingeniería directa. Una vez realizado este proceso la nueva base de datos es creada conteniendo las tablas, relaciones, atributos y claves primarias y foráneas indicadas en el diseño.

4 Instituto Superior privado peruano de sistemas SISE (2011). «Modelamiento y Diseño de Base de Datos». En: Scribd Web Site. Disponible en <http://es.scribd.com/doc/40031583/70/INGENIERIA-DIRECTA-REVERSIVA>

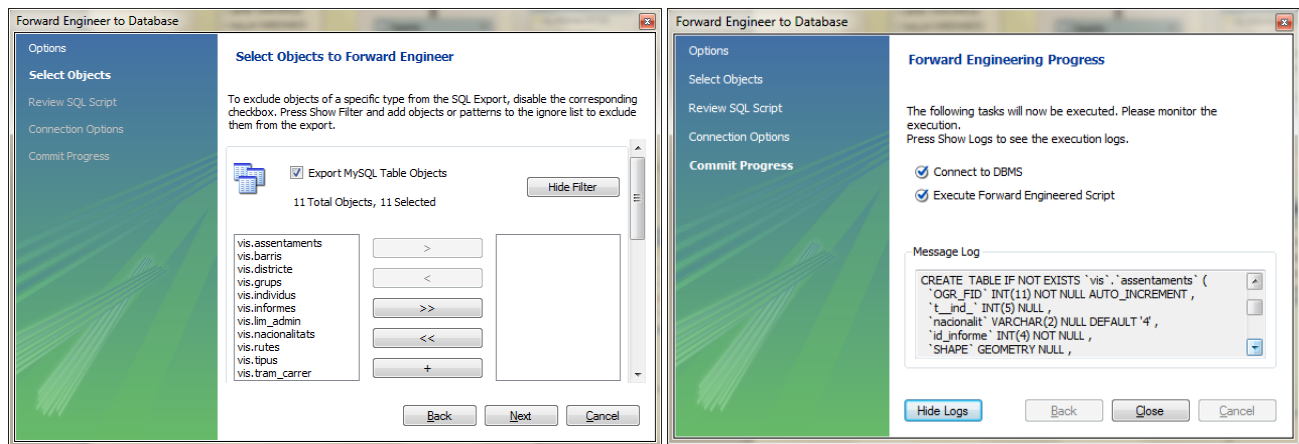


Figura 3. 1 Creación BD Ingeniería Directa -Workbench.

Como es posible recordar, en el capítulo anterior revisamos el diseño lógico de la base de datos (sección 2.1.2), en el cual incluimos en nuestras entidades el atributo SHAPE que será el encargado de almacenar los elementos espaciales. Este atributo es de tipo GEOMETRY, tipo de dato propio de MySQL para el almacenamiento de datos espaciales. GEOMETRY puede almacenar los valores de la geometría de cualquier tipo. Los otros tipos de datos espaciales simples (POINT, LINESTRING y POLYGON) restringen sus valores a un solo tipo de geometría en particular.

Nuestra base de datos necesita saber de alguna forma cuáles de las tablas creadas contienen información espacial, que tipo de elementos almacenan y cuál es su sistema de referencia. Por ello para finalizar con la creación correcta de la base de datos es necesaria la utilización de la herramienta FWtools, debido a que está nos ayudará a crear tablas vinculantes de las tablas con información espacial.

Utilizaremos el comando Ogr2Ogr que nos permite convertir de una fuente de datos Ogr a otra fuente de datos Ogr; éste soporta múltiples formatos de datos como: ESRI Shapefile, MapInfo Tab file, TIGER, GML, KML, Interlis, SQLite, ODBC, ESRI GeoDatabase (MDB format), PostGIS/PostgreSQL, MySQL. Esta vez aplicaremos el comando para sobrescribir una de las tablas que creamos anteriormente a través de la ingeniería directa. Cuando el comando se ejecute sobrescribirá nuestra tabla y creará las tablas vinculantes: geometry_columns y spatial_ref_sys e insertará la información de la tabla que fue sobrescrita. En los siguientes gráficos se muestra el comando ejecutado y las tablas resultantes.

```
Ogr2Ogr -f "MySQL" MySQL:"vis2, user=root, host = localhost,
password = xxx" -lco engine=MYISAM ruta_shape\DIST_region.shp
-nln districte -overwrite
```

Tabla geometry_columns

F_TABLE_CATALOG	F_TABLE_SCHEMA	F_TABLE_NAME	F_GEOMETRY_COLUMN	COORD_DIMENSION	SRID	TYPE
NULL	NULL	districte	SHAPE	2	1	POLYGON

Tabla spatial_ref_sys

SRID	AUTH_NAME	AUTH_SRID	SRTTEXT
1	NULL	NULL	PROJCS["ED_1950_UTM_Zone_31N",GEOGCS["GCS_European...

Figura 3. 2 Tablas geometry_columns y spatial_ref_sys.

3.2. CONFIGURACIÓN Y PRUEBAS DEL SERVIDOR DE MAPAS.

3.2.1. CONFIGURACIÓN DE MAPSERVER.

Luego de instalado el servidor (ver guía de instalación en el Anexo A) es importante revisar las extensiones disponibles con el objetivo de verificar que contemos con las necesarias para el proyecto, en este caso shape y mysql. Para ello utilizamos la aplicación mapserv.exe con el comando `ogrinfo -formats`.

```
MS4W-Shell
C:\Documents and Settings\Progress\Escritorio>ogrinfo --formats
Supported Formats:
-> "ESRI Shapefile" (read/write)
-> "MapInfo File" (read/write)
-> "UK .NTF" (readonly)
-> "SDTS" (readonly)
-> "TIGER" (read/write)
-> "SS7" (read/write)
-> "DGN" (read/write)
-> "VRT" (readonly)
-> "REC" (readonly)
-> "Memory" (read/write)
-> "BNA" (read/write)
-> "CSV" (read/write)
-> "GML" (read/write)
-> "GPX" (read/write)
-> "KML" (read/write)
-> "GeoJSON" (read/write)
-> "GMT" (read/write)
-> "SQLite" (read/write)
-> "ODBC" (read/write)
-> "PGeo" (readonly)
-> "PostgreSQL" (read/write)
-> "MySQL" (read/write)
-> "XPlane" (readonly)
-> "AVCBin" (readonly)
-> "AVCE00" (readonly)
-> "Geoconcept" (read/write)
```

Figura 3. 3 Verificación de extensiones - MapServer.

A continuación comprobamos la conexión de MapServer con nuestra base de datos en MySQL a través del comando `ogrinfo MYSQL`

```
MS4W-Shell
C:\>ogrinfo MySQL:vis,user=root,fxxxx |=PROG3SS,port=3306
INFO: Open of `MySQL:vis,user=root,fxxxx |=PROG3SS,port=3306'
      using driver `MySQL' successful.
```

Figura 3. 4 Verificación conexión MapServer - MySQL.

3.2.2. ELABORACIÓN DE ARCHIVOS DEL SERVIDOR DE MAPAS.

Para elaborar los archivos que utilizará el servidor de mapas se ha utilizado la ayuda del aplicativo cliente Quantum GIS y su herramienta de exportación de MapServer. Esta herramienta nos permite obtener un archivo en formato mapfile (extensión .map) con sentencias que pueden ser reconocidas por el servidor y con llamadas a los datos espaciales, que en nuestro caso se encuentran en la base de datos espacial de MySQL. En la siguiente figura se muestra la utilización de la herramienta de exportación.

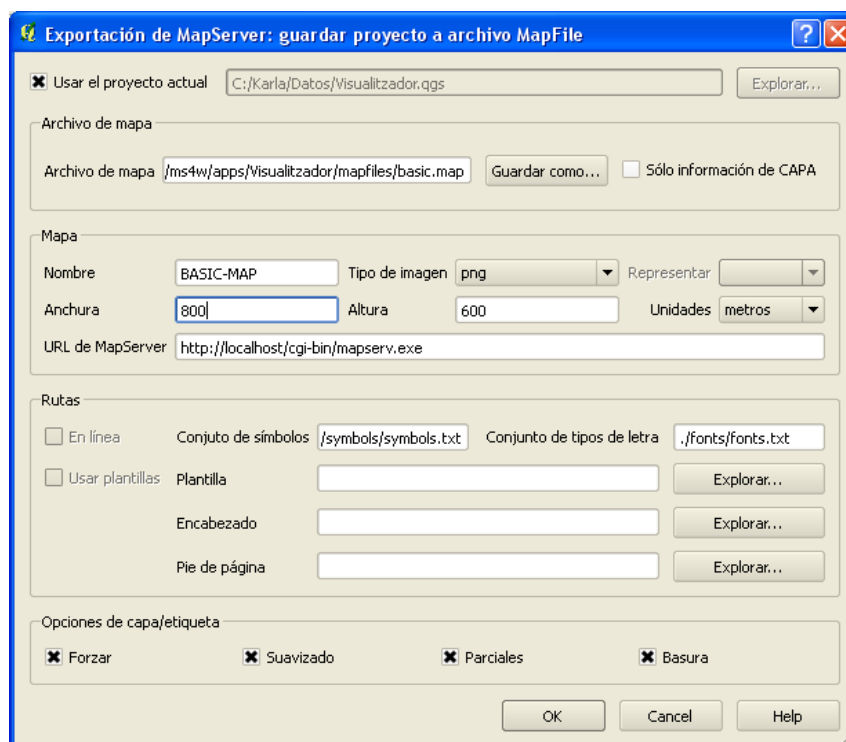


Figura 3. 5 Exportación a MapServer con Quantum GIS.

A fin de aligerar las posibles peticiones de información generaremos dos archivos .map, el primero para las capas de divisiones administrativas y callejero; y el segundo para las capas de información temática.

Una vez obtenido los mapfiles se procede a realizar los cambios

necesarios para que el servidor de mapas pueda entregar los datos que requiramos a través del uso de estos archivos.

1. Añadimos la línea de configuración CONFIG PROJ_LIB, para definir la ubicación de los archivos de proyección de la librería Proj.4. Al colocar la clave PROJ_LIB con el valor de la ubicación de nuestros archivos 'epsg' forzamos a la librería, incluida en Mapserver, a usar este valor. Los archivos 'epsg' son usados por MapServer para referenciar varios parámetros de proyección.

```
CONFIG PROJ_LIB 'MapServer_path/proj/nad'
```

2. Como en nuestro caso tenemos capas en diferentes sistemas de proyección (EPSG:900913 para la capa base del municipio de Barcelona y EPSG:23031 para todas las demás) necesitaremos definir la proyección de tal manera que MapServer pueda reproyectar los datos de las capas al sistema de referencia espacial del mapa durante el dibujado de las capas. Para ello modificamos el parámetro PROJECTION de nuestros archivos map.

```
PROJECTION
  "init = epsg:23031"
END
```

3. A nivel de capas es necesario redefinir la sección METADATA en la cual es posible colocar los parámetros que serán usados por los servicios Open Geospatial Consortium (OGC). En nuestro caso los servicios con los cuales trabajaremos serán Web Mapping Service (WMS), Web Feature Services (WFS) y el lenguaje de marcado geográfico GML para leer los elementos espaciales. Si no configuramos estos parámetros los servicios mencionados no podrán encontrar la información que requeriremos a través de las peticiones al servidor de mapas. En el siguiente cuadro de muestra un ejemplo de la definición de METADATA para una de las capas de información temática.

```
METADATA
  'ows_title' 'assentaments'
  "wfs_title" "assentaments"
  "wfs_typename" "assentaments"
  "wms_title" "assentaments"
  "gml_featureid" "assentaments"
  "gml_include_items" "all"
  "gml_exclude_items" "nacionalit"
END
```

4. Ha sido necesario también modificar la cadena de conexión a la base de datos espacial por una de tipo OGR para poder conectarnos a nuestra base de datos MySQL. A continuación se muestra un ejemplo de la modificación:

```
#sentencia anterior DATA 'MySQL:vis,  
host=localhost,port=3306,user=root,password=xxx'  
  
CONNECTIONTYPE OGR  
CONNECTION 'MySQL:  
vis,user=root,password=xxx,port=3306,tables=assentaments'  
DATA "select a.SHAPE, a.t__ind_, a.nacionalit,  
a.id_informe, n.nacionalitat, i.data from assentaments as  
a, nacionalitats as n, informes as i where a.id_informe =  
i.id_informe and a.nacionalit = n.codi and n.id_tipus = 3"
```

3.2.3. PRUEBAS DEL SERVIDOR DE MAPAS

Para comprobar que la configuración sea correcta se realizarán dos pequeñas pruebas con los servicios WMS y WFS del servidor de mapas:

1. Llamada WMS para la capa de límites administrativos que es usada como capa base del mapa.

[http://localhost/cgi-](http://localhost/cgi-bin/mapserv.exe?MAP=C:/ms4w/apps/Visualitzador/mapfiles/basic.map&LAYER=S=lim_admin&FORMAT=image/png&UNITS=m&SERVICE=WMS&VERSION=1.1.1&REQUEST=GetMap&STYLES=&SRS=EPSG%3A900913&BBOX=210306.31001902,5058381.9612281,268895.28998098,5082001.0020737&WIDTH=1533&HEIGHT=618)

[bin/mapserv.exe?MAP=C:/ms4w/apps/Visualitzador/mapfiles/basic.map&LAYER=S=lim_admin&FORMAT=image/png&UNITS=m&SERVICE=WMS&VERSION=1.1.1&REQUEST=GetMap&STYLES=&SRS=EPSG%3A900913&BBOX=210306.31001902,5058381.9612281,268895.28998098,5082001.0020737&WIDTH=1533&HEIGHT=618](http://localhost/cgi-bin/mapserv.exe?MAP=C:/ms4w/apps/Visualitzador/mapfiles/basic.map&LAYER=S=lim_admin&FORMAT=image/png&UNITS=m&SERVICE=WMS&VERSION=1.1.1&REQUEST=GetMap&STYLES=&SRS=EPSG%3A900913&BBOX=210306.31001902,5058381.9612281,268895.28998098,5082001.0020737&WIDTH=1533&HEIGHT=618)

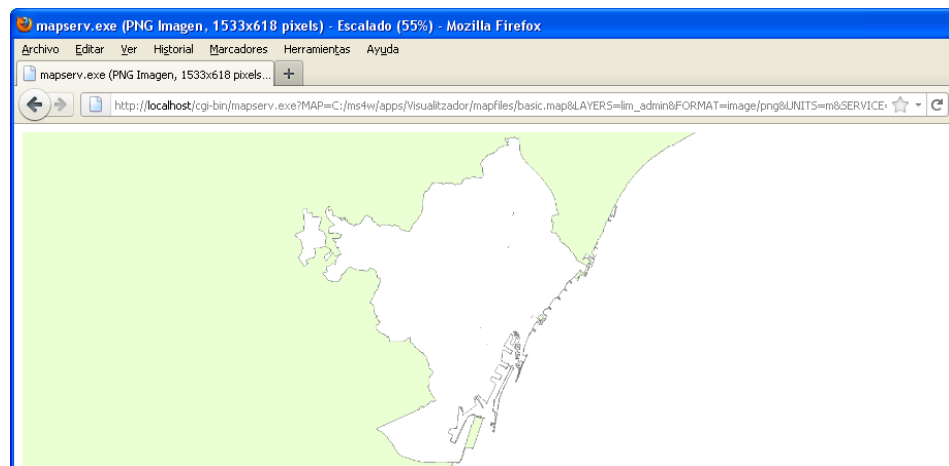


Figura 3. 6 Prueba servicio WMS servidor de mapas.

2. Llamada WFS de la capa temática grups, cuyo respuesta es observada utilizando la herramienta para el desarrollo de aplicaciones web firebug del navegador web Firefox.

[http://localhost/cgi-](http://localhost/cgi-bin/mapserv.exe?map=C:/ms4w/apps/Visualitzador/mapfiles/tematic.map&S=grups&FORMAT=application/json&SERVICE=WFS&VERSION=1.1.0&REQUEST=GetFeature&TYPENAME=grups)

[bin/mapserv.exe?map=C:/ms4w/apps/Visualitzador/mapfiles/tematic.map&S=grups&FORMAT=application/json&SERVICE=WFS&VERSION=1.1.0&REQUEST=GetFeature&TYPENAME=grups](http://localhost/cgi-bin/mapserv.exe?map=C:/ms4w/apps/Visualitzador/mapfiles/tematic.map&S=grups&FORMAT=application/json&SERVICE=WFS&VERSION=1.1.0&REQUEST=GetFeature&TYPENAME=grups)

RVICE=WFS&VERSION=1.0.0&REQUEST=DescribeFeatureType&TYPE
NAME=feature:grups&OUTPUTFORMAT=XMLSCHEMA

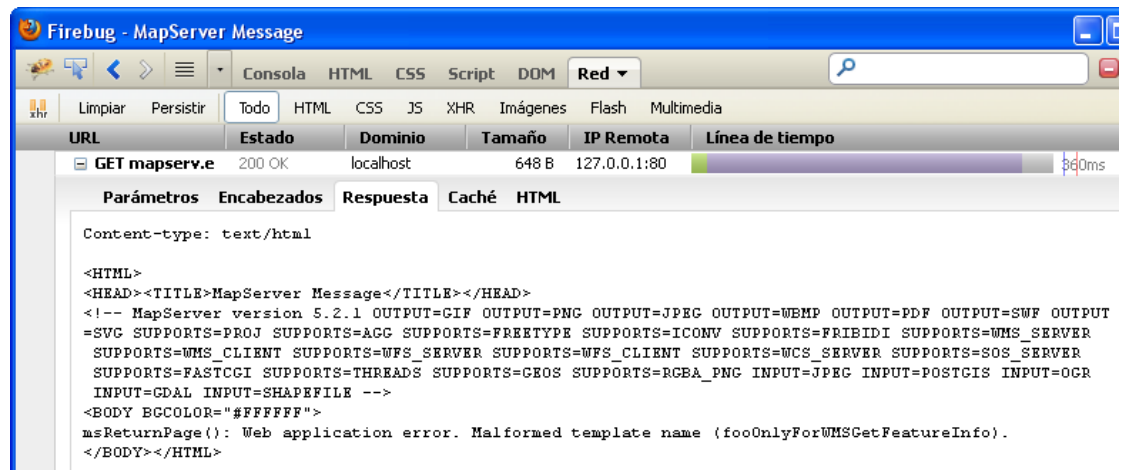


Figura 3. 7 Prueba servicio WFS servidor de mapas.

3.3. DESARROLLO E IMPLEMENTACIÓN DE LA APLICACIÓN CLIENTE, VISOR DE MAPAS.

La aplicación visualizador de mapas será la encargada de relacionar las otras partes del sistema (servidor de mapas, base de datos, servidor web).

3.3.1. DESCRIPCIÓN DE LA METODOLOGÍA DE DESARROLLO UTILIZADA.

Para la implementación de la aplicación cliente se utilizó el modelo de desarrollo de software por prototipos.

El Modelo de prototipos, pertenece a los modelos de desarrollo evolutivo. El prototipo debe ser construido en poco tiempo, usando los programas adecuados y no se debe utilizar mucho dinero pues a partir de que éste sea aprobado nosotros podemos iniciar el verdadero desarrollo del software. El diseño rápido se centra en una representación de aquellos aspectos del software que serán visibles para el cliente o el usuario final. Este diseño conduce a la construcción de un prototipo, el cual es evaluado por el cliente para una retroalimentación; gracias a ésta se refinan los requisitos del software que se desarrollará. La interacción ocurre cuando el prototipo se ajusta para satisfacer las necesidades del cliente. Esto permite que al mismo tiempo el desarrollador entienda mejor lo que se debe hacer y el cliente vea resultados a corto plazo.

En nuestro caso durante la realización de la aplicación se crearon varios prototipos que fueron paulatinamente mostrados al tutor de la empresa para recoger su opinión y retroalimentación.

3.3.2. ESTRUCTURA Y DESARROLLO DE LA APLICACIÓN.

Para la implementación de la aplicación cliente (visualizador de mapas) se ha decidido dividir la lógica de la programación en 5 módulos que se describen a continuación:

Módulo control de datos: Este pequeño módulo escrito en PHP nos ayuda a cargar datos alfanuméricos desde la base de datos de MySQL, por ejemplo se encarga de obtener la lista de los distritos y los barrios para poder realizar la localización, también nos ayuda a traer los atributos de la información temática y las fechas de los reportes para realizar el filtrado de ésta información.

Módulo Parámetros: Se encarga de gestionar aquellas opciones del sistema que son parametrizables como por ejemplo los mensajes que muestra la aplicación al usuario, las rutas de archivos, rutas del servidor de mapas, nombres de las capas, rutas de imágenes, etc.

Módulo Capas: Contiene la definición de cada una de las capas que se podrán visualizar en la aplicación, describiendo para cada una de ellas si es capa base o sobrepuesta, si es tipo imagen o elemento, así como el estilo y el filtro inicial para las capas de información temática. Este módulo se comunica con el de parámetros para obtener los datos del servidor de mapa y los nombres de las capas según los archivos .map.

También es el que se comunica directamente con el servidor de mapas al hacer las peticiones de las capas de información. En este sentido utiliza tres tipos de llamadas (tres tipos de servicios) para realizar las peticiones.

1. Servicio WMS para obtener las capas base del municipio de Barcelona, las capas de divisiones administrativas y las del callejero.

2. Servicio WFS para obtener capas de tipo vector de la información temática que utiliza la aplicación, utilizando el lenguaje GML para describir los elementos.
3. Servicio de capas de google para obtener las capas base de Google Satellite y Google Street.

Módulo Herramientas: Este módulo maneja casi todas las actividades que podemos realizar sobre el mapa, en la aplicación de visualización. Es decir implementa las funcionalidades de las herramientas de navegación, de historial, de medición, de localización, de impresión, de identificación y la herramienta de selección de mapas bases.

Interactúa con el módulo de capas para acceder a las capas que se visualizan sobre el mapa, con el módulo geoportal para acceder al objeto mapa sobre el cual se realizarán las funcionalidades y con el módulo control de datos para obtener los datos alfanuméricos que necesite.

Módulo Geoportal: Es el módulo central pues interactúa con todos los demás y contiene el mapa, objeto principal de la aplicación, el cual se halla ubicado dentro de un panel (Objeto MapPanel de GeoExt). Aquí se encuentra la implementación del árbol de capas y del filtrado de la información temática, funcionalidad más importante de la aplicación visualizador. Además en este módulo se define el diseño y la disposición de las diferentes partes de la interfaz de la aplicación.

Utiliza el módulo de parámetros para obtener el texto por defecto de las diferentes opciones de la aplicación, como por ejemplo el texto de los combos vacíos, así como también para obtener los nombres de las capas según los archivos map.

Interactúa con el módulo de capas para definir el conjunto de información espacial que se podrá visualizar en el mapa. Por otro lado interactúa con el módulo herramientas colocando las funcionalidades en la barra de herramientas del panel del mapa y para obtener el contenido del panel de identificación de elementos.

Y utiliza el módulo de control de datos para obtener la información que se mostrarán en el formulario de filtrado.

Módulo inicial: Se encarga de convocar a todos los módulos anteriores, a las librerías javascript necesarias para el funcionamiento de éstos y a los archivos de estilo que utiliza la interfaz de la aplicación.

Cada uno de estos módulos se encuentra escrito en un archivo físico y son parte principal del conjunto de archivos que permiten el funcionamiento de la aplicación, el cual se detalla en el Anexo A de este documento. La siguiente figura esquematiza la interacción entre los diferentes módulos de la aplicación.

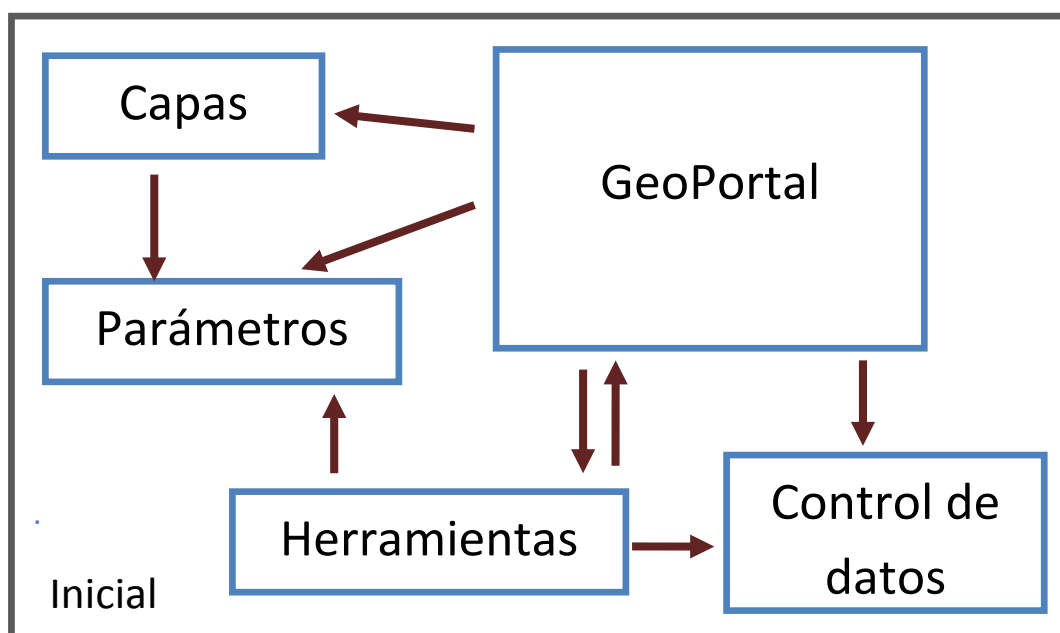


Figura 3. 8 Módulos Aplicación Web

3.4. IMPLEMENTACIÓN APLICACIÓN DE ACTUALIZACIÓN DE DATOS.

Para que el sistema realizado pueda ser de utilidad para la empresa es necesario que la información temática territorial pueda ser actualizada, por ello se ha realizado pequeña aplicación de escritorio que servirá para que sea posible dar mantenimiento a los datos que presenta el visualizador de mapas. Los datos que esta aplicación permitirá actualizar son los de los informes mensuales que deben presentar los educadores de calle en cuanto a: individuos sin techo, grupos y asentamientos de personas sin techo y las rutas seguidas por los educadores.

En cuanto a la plataforma tecnológica, la aplicación ha sido construida utilizando el lenguaje de programación java y el entorno de de desarrollo

integrado Netbeans, ambos programas gratuitos. Funcionalmente permite crear nuevos informes, asociando a ellos un grupo de elementos espaciales y eliminar informes junto a sus elementos espaciales relacionados, para lo cual ha sido necesario crear tres procedimientos almacenados en la base de datos (`getInforme`, `newInforme`, `deleteInforme`) que nos ayudarán a realizar las operaciones sobre los datos de manera más segura. Como aportación de interés, a continuación se detalla el proceso de implementación seguido para realizar ambas funcionalidades.

Para crear un nuevo informe:

1. Se debe verificar que no exista un informe para el mismo mes-año seleccionado por el usuario. Para ello utilizamos el procedimiento almacenado **getInforme** que dado un tipo de información temática, un mes y un año nos retornara el id del informe en caso de que este ya existiera.
2. Si no existe informe para la fecha seleccionada procedemos a transformar los datos de shape a MySQL ejecutando el comando de conversión de datos `ogr2ogr` de la librería `Gdal`.

```
command ="ogr2ogr -f
\"MySQL\"MySQL:\"vis,user=root,host=localhost,password=xxx\" \"
+ pathFile + \" -nln \"
+ nameType + \" -update -append\";
```

Donde `pathfile` es la ruta completa del archivo shape y `nameType` es el nombre de la tabla espacial almacenada en MySQL en la que deseamos almacenar los datos que se encuentran en formato shape.

3. Con la ayuda del procedimiento almacenado **newInforme**, creamos un nuevo registro en la tabla `informes` con la información con el tipo de información temática, mes y año del reporte. Luego de crear el registro el procedimiento también se encargará de actualizar el campo `id_informe` de todos los datos que fueron ingresados en el paso 2 para que pueda existir relación entre el informe y los datos espaciales de las tablas de información temática.

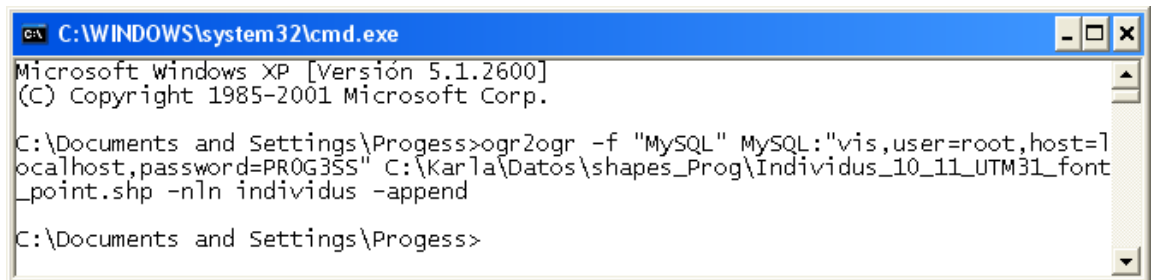
Para la eliminación de informes se han realizado los siguientes pasos:

1. A través del procedimiento almacenado **deleteInforme** se selecciona la tabla espacial de información temática que corresponda a la del tipo del

informe a eliminar y luego se borra dentro de esta tabla los registros cuyo identificador de informe corresponda con el del informe a eliminar.

2. Una vez eliminado los registros que se relacionaban al informe a eliminar a través de una clave foránea, el procedimiento de eliminación podrá realizar la eliminación del registro correspondiente en la tabla informes.

Es importante señalar que si bien se ha desarrollado la aplicación de mantenimiento a fin de facilitar la tarea de actualización de los datos, también es posible y queda a potestad de la empresa realizar esta actualización a través de otros medios, para lo cual se sugiere seguir la secuencia de los pasos indicados en esta sección utilizando los medios que establezcan convenientes. Por ejemplo es posible utilizar los mismos procedimientos almacenados que ha utilizado la aplicación desde una herramienta de administración de bases de datos MySQL (ej. PhpMyAdmin) y ejecutar los comandos de la librería Gdal directamente desde la consola de comandos del sistema operativo, como lo muestra el siguiente gráfico.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Progress>ogr2ogr -f "MySQL" MySQL:"vis,user=root,host=localhost,password=PROG355" C:\Karla\Datos\shapes_Prog\Individus_10_11_UTM31_font_point.shp -nln individuos -append

C:\Documents and Settings\Progress>
```

Figura 3. 9 Ejecución comando de transformación de datos.

CAPÍTULO IV

4. ELABORACIÓN Y CARGA DE DATOS

4.1. TRANSFORMACIÓN Y CARGA DE DATOS ESPACIALES

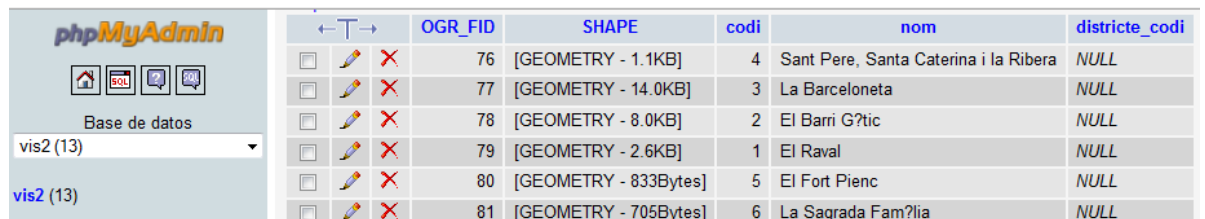
Para la transformación de formatos y carga de los datos de entrada (archivos shape) ha sido necesaria la utilización de las herramientas FWtools. A través de la consola de comandos de FWtools realizamos la ejecución de:

```
ogr2ogr -f "MySQL" MySQL:" vis, user=root, host=localhost, password=xxx" ruta_shape\barris.shp -nln barris -append
```

El cual da por resultado el ingreso de la información del archivo barris.shp a la tabla barris de la base de datos. Además actualiza las tablas geometry_columns y spatial_ref_sys con la información de la tabla recientemente actualizada.

Luego es importante verificar que la transformación y carga de datos se haya realizado exitosamente. Esta comprobación se puede realizar de tres maneras:

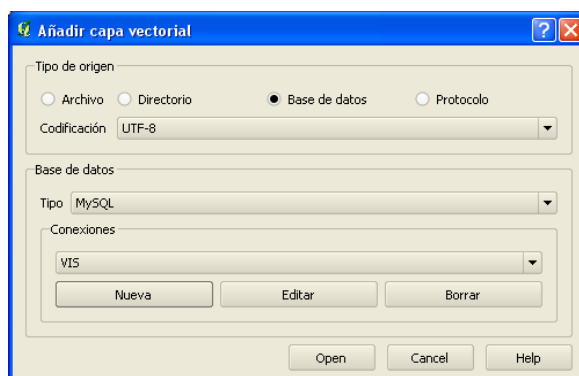
1. Utilizando phpMyAdmin. Comprobamos que los datos se hayan insertado



OGR_FID	SHAPE	codi	nom	districte_codi
76	[GEOMETRY - 1.1KB]	4	Sant Pere, Santa Caterina i la Ribera	NULL
77	[GEOMETRY - 14.0KB]	3	La Barceloneta	NULL
78	[GEOMETRY - 8.0KB]	2	El Barri Gòtic	NULL
79	[GEOMETRY - 2.6KB]	1	El Raval	NULL
80	[GEOMETRY - 833Bytes]	5	El Fort Pienc	NULL
81	[GEOMETRY - 705Bytes]	6	La Sagrada Família	NULL

Figura 4. 1 PHPMyAdmin - comprobación de datos.

2. Utilizando un cliente Gis, en este caso Quantum gis, en el cual añadimos la capa vectorial que tenga por origen la base de datos de MySQL.



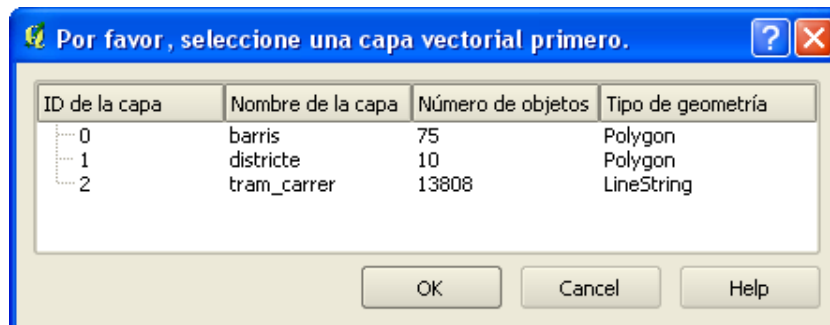
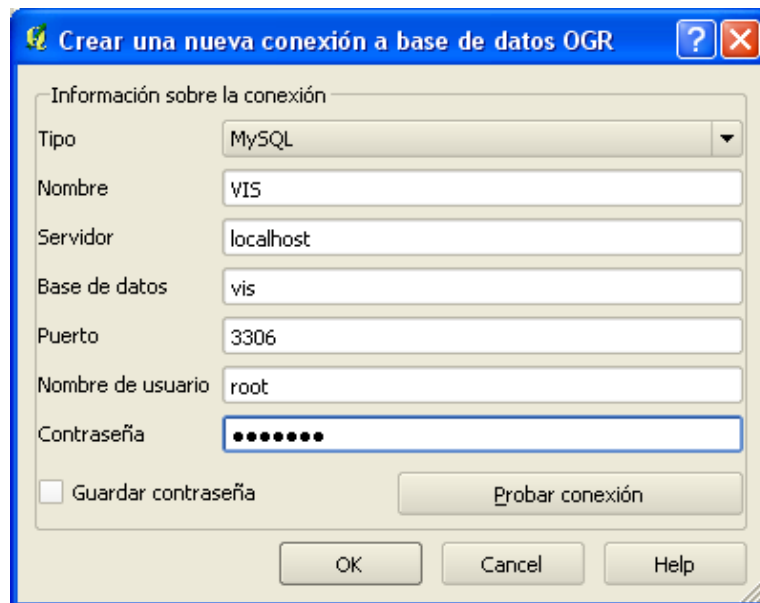


Figura 4. 2 Comprobación de datos - Quantum Gis.

3. Revisión desde la consola de mapserv.exe. Ahora comprobamos que MapServer lo reconozca como un archivo espacial a través de comandos en el shell del servidor.

```
ogrinfo MySQL:vis, user=root, password=xxx, port=3306 barris
-summary
```

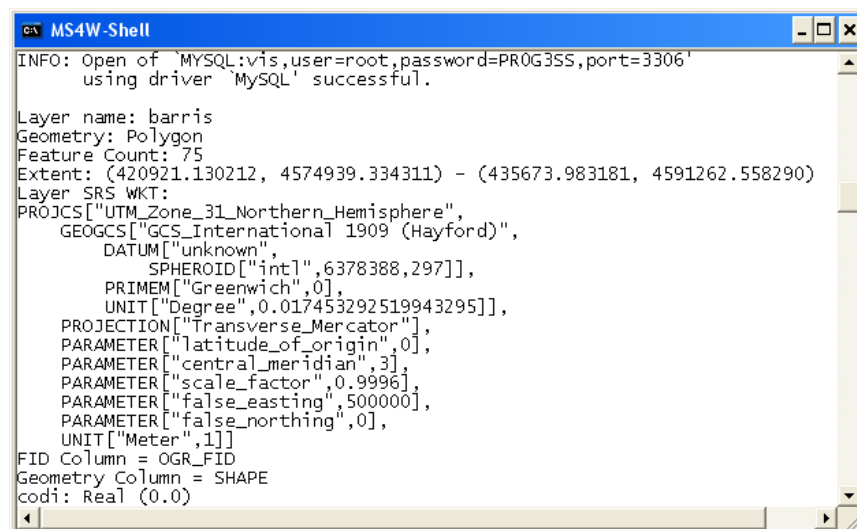


Figura 4. 3 Comprobación de datos - MapServer.

4.2. ELABORACIÓN Y CARGA DE DATOS ALFANUMÉRICOS

- La información de las tablas alfanuméricas tipus, nacionalitats ha sido ingresada de manera manual debido a que el número de registro son pocos.
- Los campos alfanuméricos de las tablas con representación espacial se han ingresado en su mayoría de manera automática pues acompañaban a las tablas relacionadas al archivo shape.

CAPÍTULO V

5. EXPLOTACIÓN Y PRUEBAS DEL SISTEMA

5.1. PRESENTACIÓN DE RESULTADOS.

Para evaluar los resultados nos guiaremos de los criterios establecidos en los requerimientos funcionales del sistema, descritos en el primer capítulo.

EXPLOTACIÓN 1

Funcionalidad:

El sistema permitirá mostrar la información temática sobre tres tipos de mapas base: Google Satélite, Google callejero y distritos de Barcelona.

Resultado: Para probar esta funcionalidad utilizaremos la lista desplegable de los diferentes mapas base con los que cuenta el visualizador: Google Street, Google Satellite y los distritos de Barcelona.

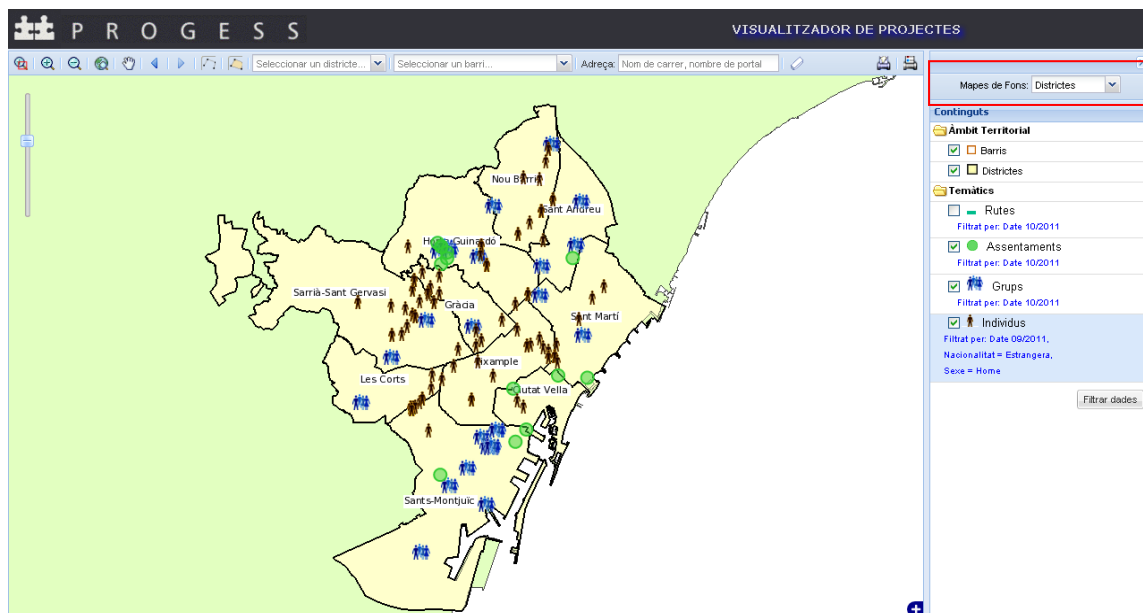


Figura 5. 1 Mapa Base Districtos

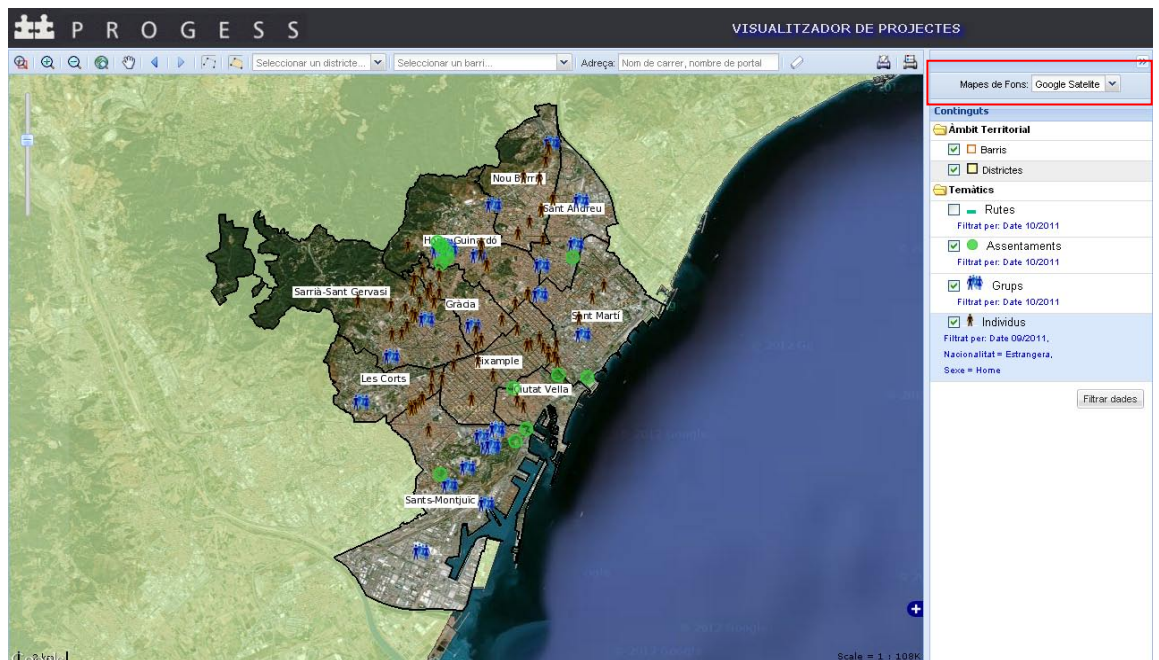


Figura 5. 2 Mapa Base Google Satellite.

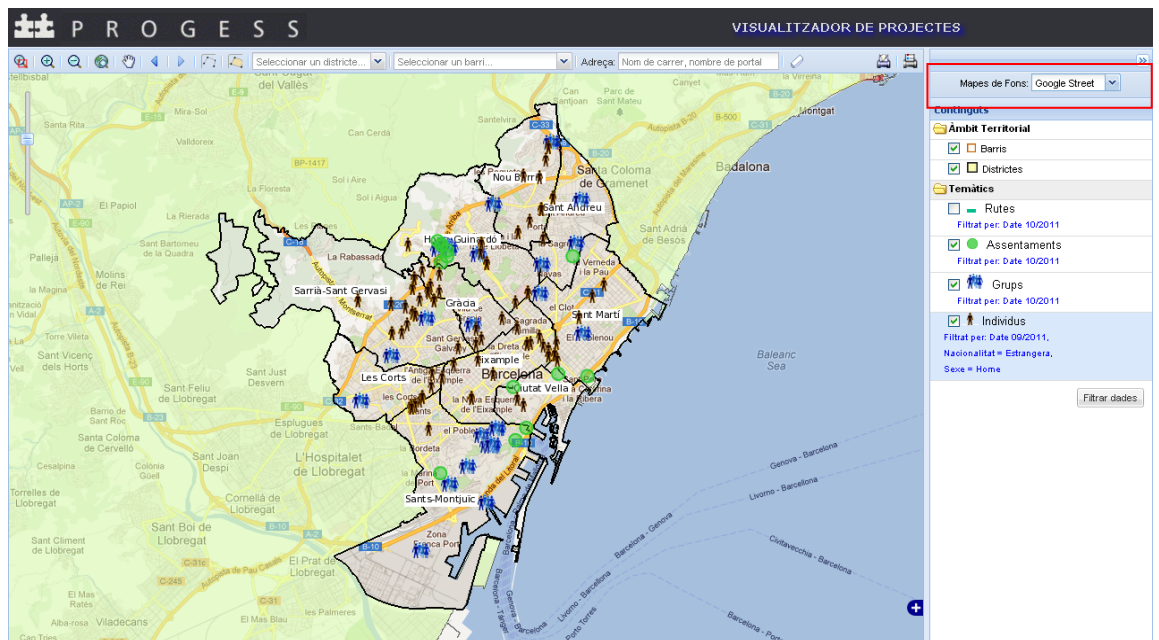


Figura 5. 3 Mapa Google Street.

EXPLLOTACIÓN 2

Funcionalidad:

El sistema deberá mostrar una simbolización amigable de las diferentes capas de información territorial.

Resultado: Para garantizar la buena simbolización de las capas se ha utilizado colores diferentes para cada una. Además las capas de información temática cuentan con simbolización por gráficos los cuales hacen referencia

a su tipo de información, como se muestra en la siguiente imagen del sistema:

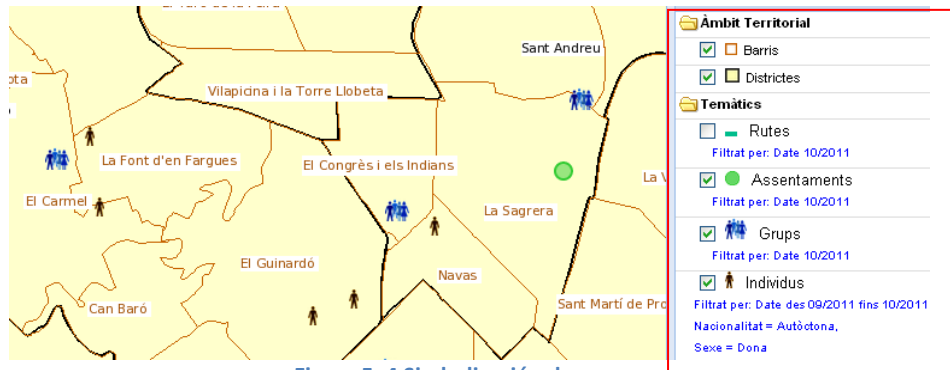


Figura 5. 4 Simbolización de capas.

EXPLOTACIÓN 3

Funcionalidad:

El sistema podrá visualizar las distintas capas de información, en función de la escala.

Resultado: Como lo muestran las siguientes imágenes la visualización de las capas varía de acuerdo al nivel de escala. Más específicamente en la primera imagen a escala 1:54K no se visualizan los barrios, mientras que en la segunda imagen a escala 1:27K éstos se visualizan. Para emplear esta opción se puede hacer uso de la barra de deslizamiento que se encuentra a un costado del mapa o utilizar la rueda del ratón.



Figura 5. 5 Mapa a escala 1:54k.

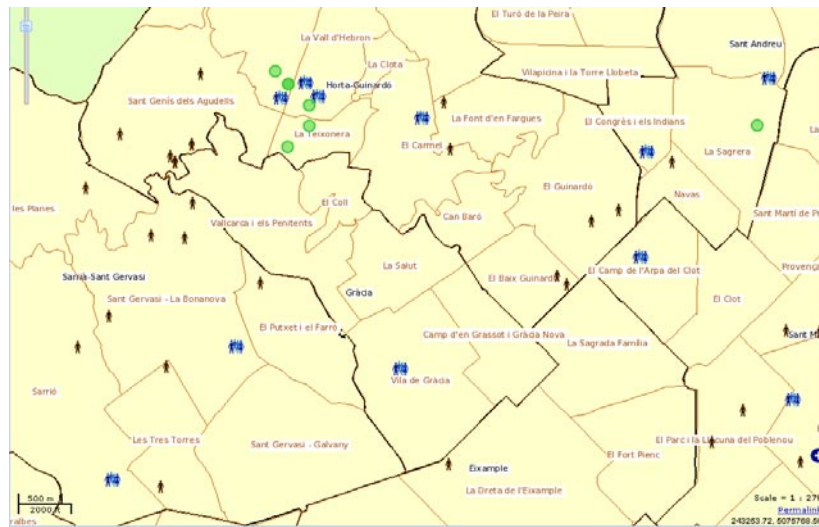


Figura 5. 6 Mapa a escala 1:27k.

EXPLORACIÓN 4

Funcionalidad:

Proveerá de las herramientas básicas de visualización (zoom in, zoom out, pan, fullextent, escala), mapa guía y visualización de coordenadas x, y.

Resultado: La aplicación cuenta con una barra de botones que contienen las herramientas de navegación más utilizadas las cuales son: zoom in, zoom out, pan y fullextent. La herramienta de escala se la ha colocado dentro del cuadro de mapa a manera de un deslizador para que el usuario pueda cambiar la escala del mapa con facilidad, también es posible modificarla utilizando la rueda del ratón cuando el deslizador está seleccionado. Además se han colocado como adicionales las herramientas para medir longitud y área. El siguiente gráfico muestra el uso de una de las herramientas citadas: zoomout.

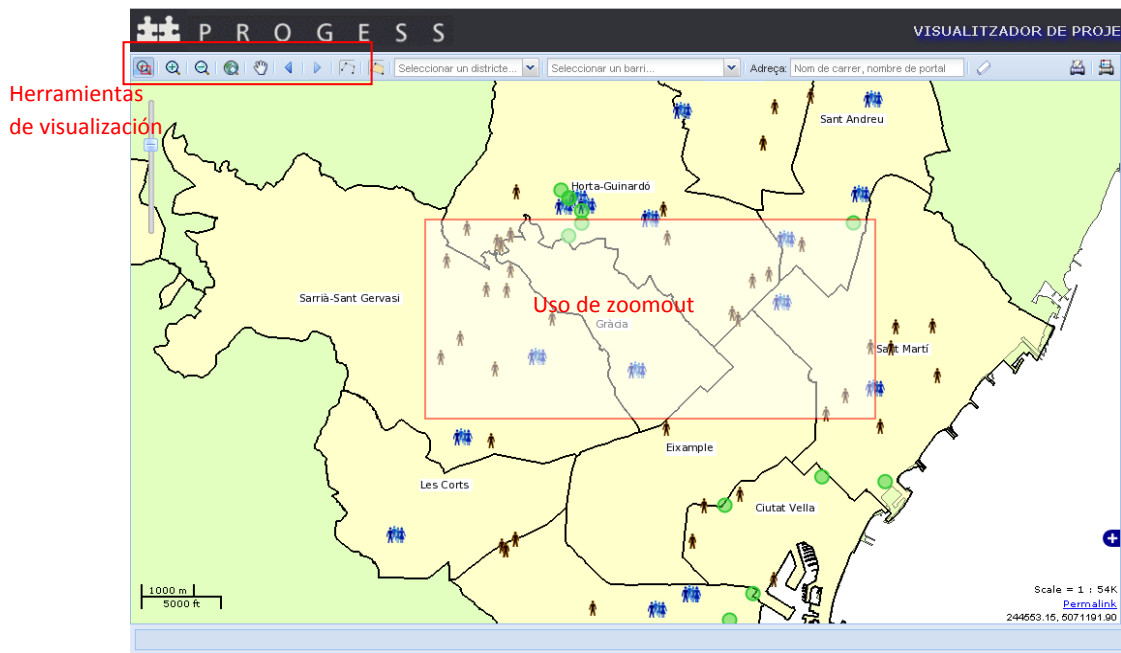


Figura 5. 7 Uso herramientas de navegación.

EXPLORACIÓN 5

Funcionalidad:

Permitirá el control de la visualización de capas.

Resultado: La lista de capas ha sido elaborada con un control tipo árbol que permite al usuario decidir si una capa debe estar visible u oculta haciendo clic en una casilla de verificación. Además del control de visualización cada capa dentro del árbol cuenta con un ícono que representa su simbolización dentro del mapa y un texto que indica los criterios de filtro aplicados sobre esa capa.

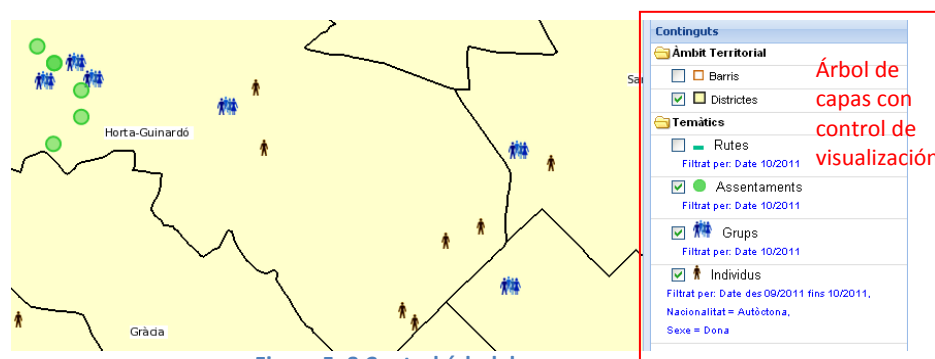
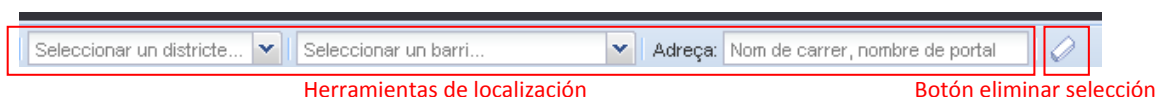


Figura 5. 8 Control árbol de capas.

Funcionalidad:

En función del ámbito territorial al que pertenecen (barrios, distritos, calles), la aplicación deberá permitir la localización de la información temática.

Resultado: Dentro de la barra de herramientas se ubican tres opciones que permiten localizar la información escogiendo entre distritos, barrios o ingresando una dirección postal. Además seguido de estas herramientas se ubica un botón que permita borrar la selección realizada.



El funcionamiento de éstas búsquedas se detallan a continuación:

Localización por distrito:

1. Se selecciona un distrito de la lista.



Figura 5. 9 Selección de distrito.

2. La aplicación hace un zoom y selección del distrito elegido. El elemento seleccionado se muestra en contorno azul para que pueda ser rápidamente reconocido por el usuario.

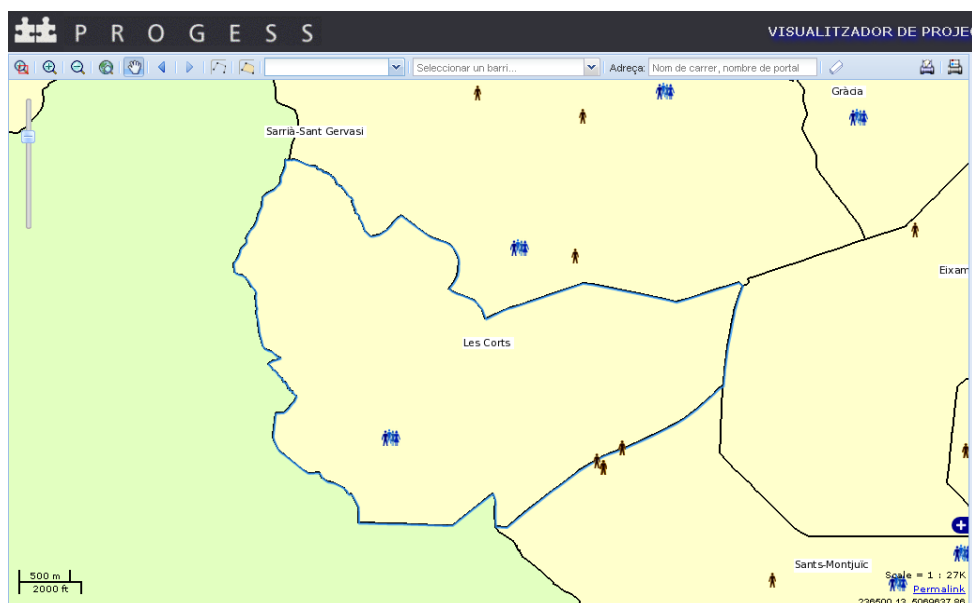


Figura 5. 10 Zoom a distrito seleccionado.

Localización por barrio:

1. Se selecciona un barrio de la lista.

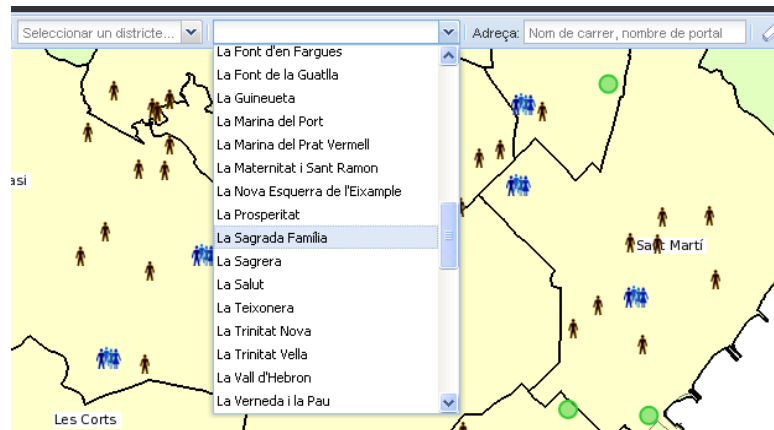


Figura 5. 11 Selección de barrio.

2. La aplicación hace un zoom y selección del barrio elegido. Al igual que en el caso anterior, el elemento seleccionado se muestra en contorno azul para que pueda ser rápidamente reconocido por el usuario.

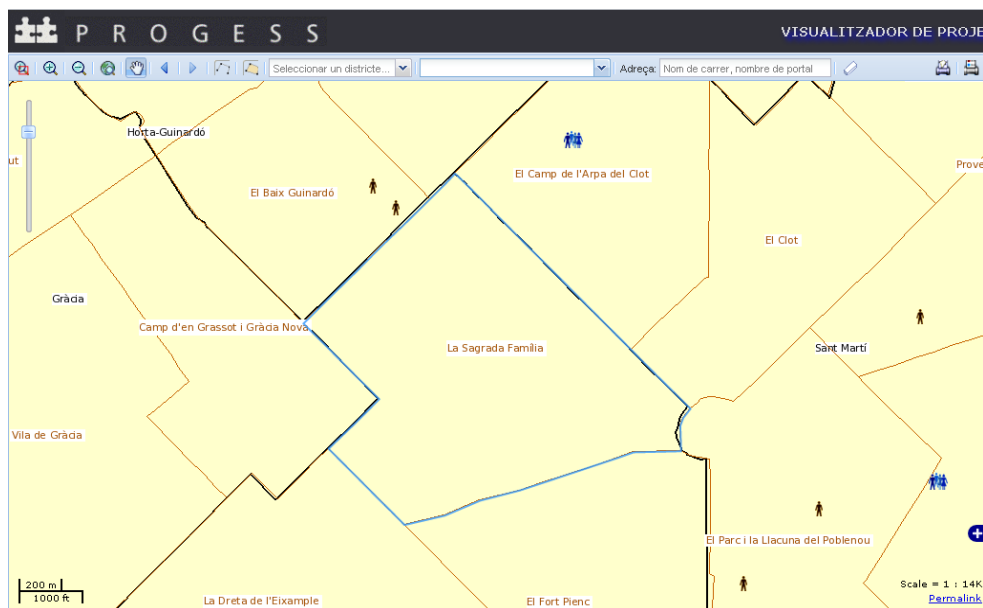


Figura 5. 12 Zoom a barrio seleccionado.

Localización por dirección calle o dirección:

1. Para realizar esta localización es necesario que el usuario ingrese su criterio de búsqueda en el formato especificado: nombre de calle o nombre de calle, número de portal.

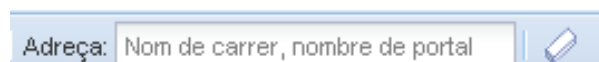


Figura 5. 13 Herramienta localización por dirección.

2. Luego de escribir el texto de búsqueda y presionar la tecla enter, la aplicación hace un zoom y selección a la dirección especificada (en caso de encontrarla). Además se realiza un cambio automático al mapa base de calles de Google para que el usuario pueda visualizar de manera más óptima el resultado de la búsqueda. En el gráfico siguiente se muestra el resultado para la dirección “Sicilia, 199”.

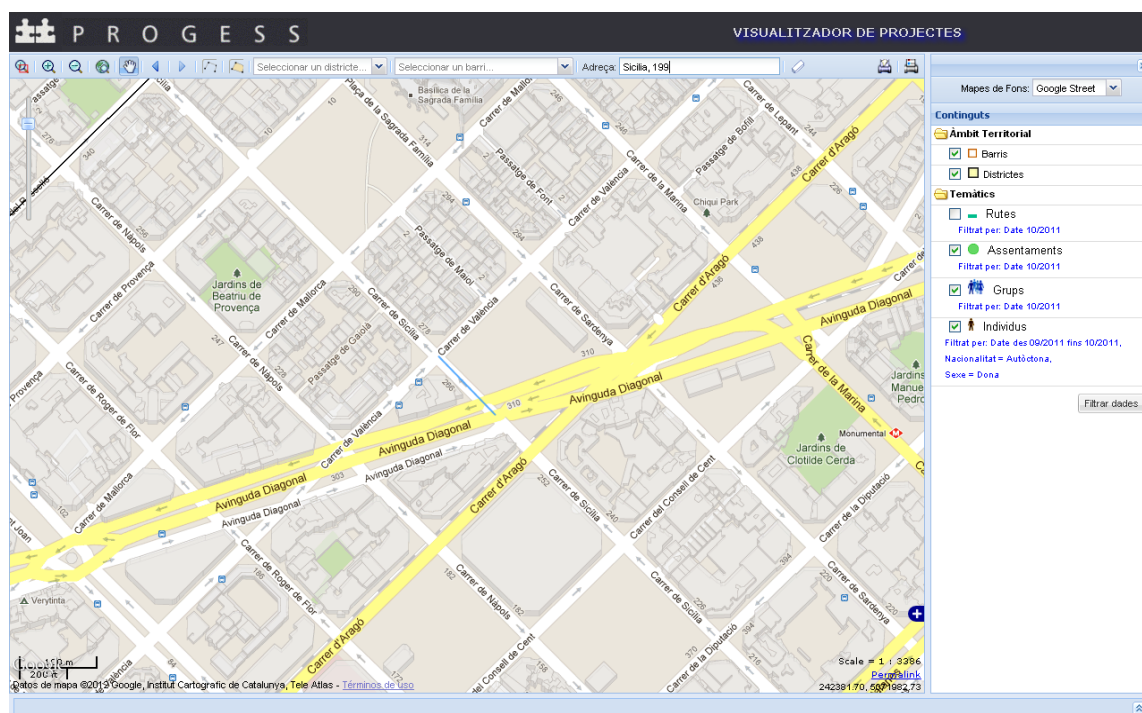


Figura 5. 14 Resultado búsqueda por calle.

EXPLORACIÓN 7

Funcionalidad:

En función de los valores de sus atributos (por ejemplo, en caso de los individuos, según sexo o según origen), la aplicación deberá permitir filtrar la información temática.

Resultado: Para poder realizar esta funcionalidad se ha incluido un formulario que permite configurar el filtro deseado en base a los siguientes parámetros: tipo de información temática, fecha inicial y final de los informes y atributos de la información temática. A continuación se detallan varios ejemplos de filtrado:

Ejemplo 1: Filtrado por individuos

Filtrar individuos cuya nacionalidad sea “Extranjera” y sean “hombres”, y que pertenezcan a los informes desde el mes de septiembre hasta el mes de octubre del 2011. Si revisamos el primer y segundo gráfico podemos ver el cambio sobre el mapa al aplicar el filtro, fíjese también en el cambio en el árbol de mapas con respecto al texto indicativo de filtrado.

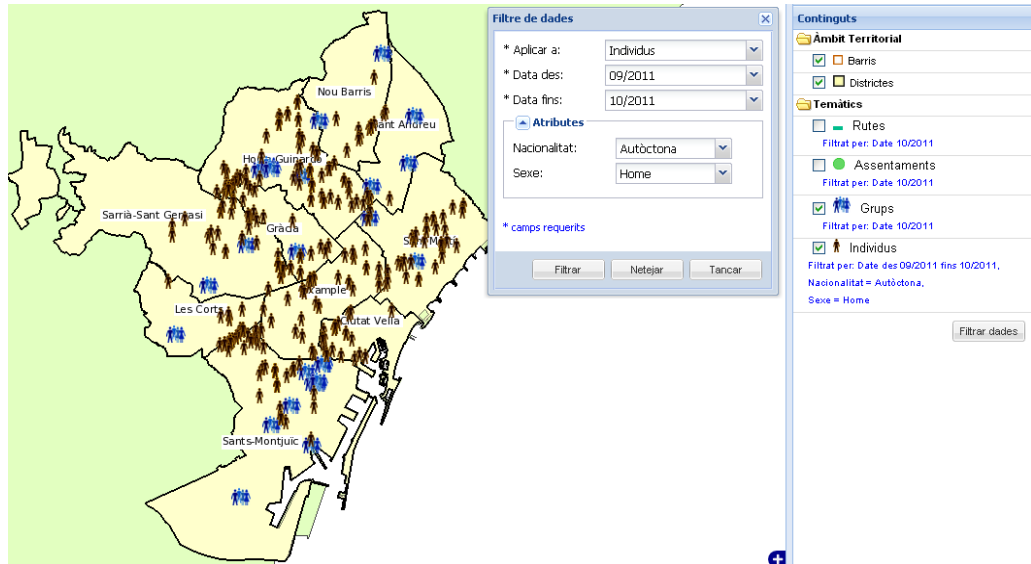


Figura 5. 15 Filtro por individuos.

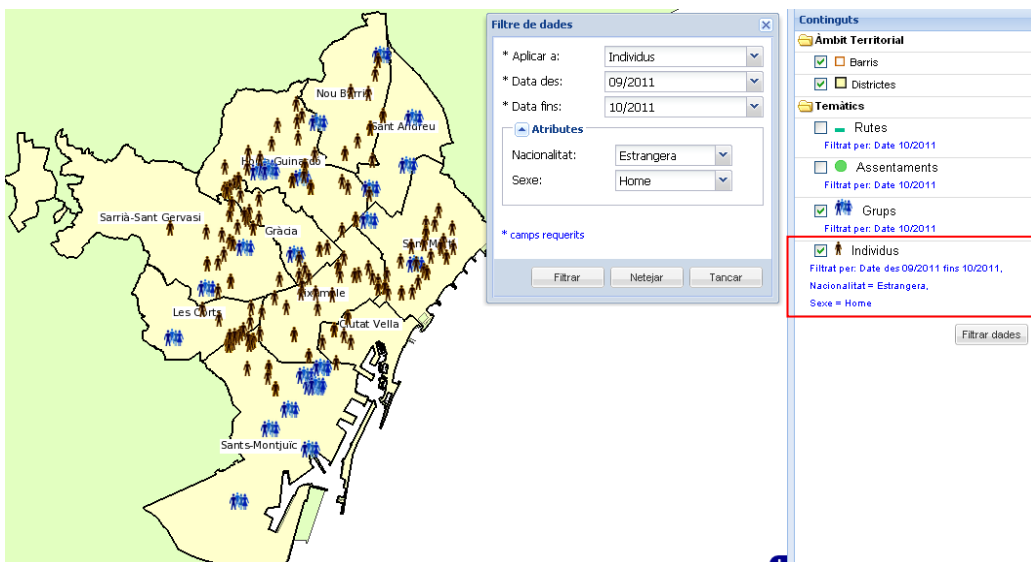


Figura 5. 16 Resultado filtro por individuos.

Ejemplo 2: Filtrado por asentamientos

Filtrar asentamientos cuya nacionalidad sea “Rumanos”, y que pertenezcan a los informes desde el mes de septiembre hasta el mes de octubre del 2011. Al igual que en el caso anterior se presentan dos imágenes que

permiten observar el cambio sobre la capa de asentamientos al aplicar el filtro. Fíjese también en que con respecto al caso anterior los atributos mostrados son diferentes pues estos dependen del tipo de información temática a la que se le aplicará el filtro.

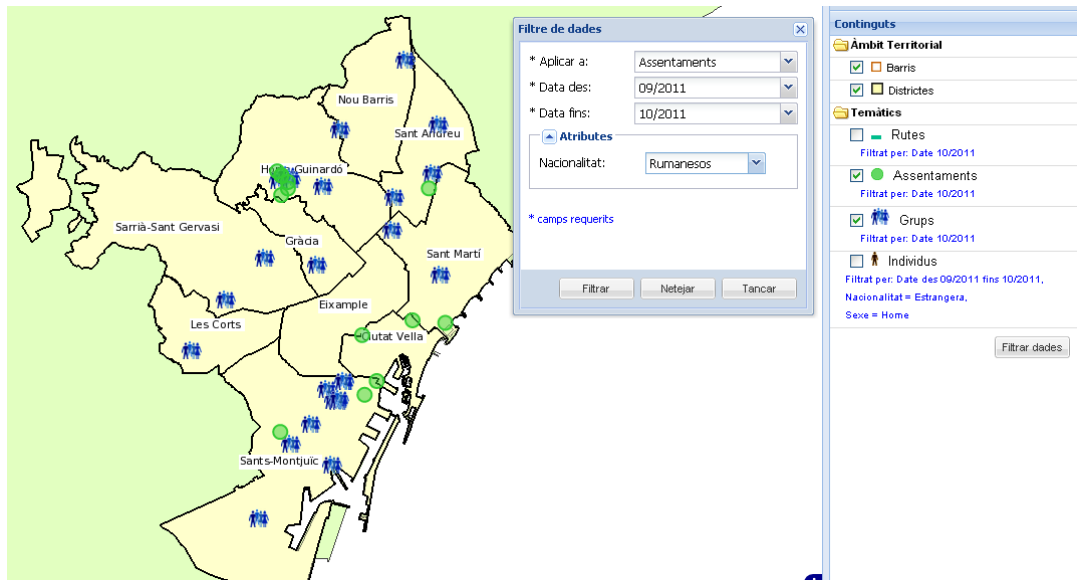


Figura 5. 17 Filtro por asentamientos.

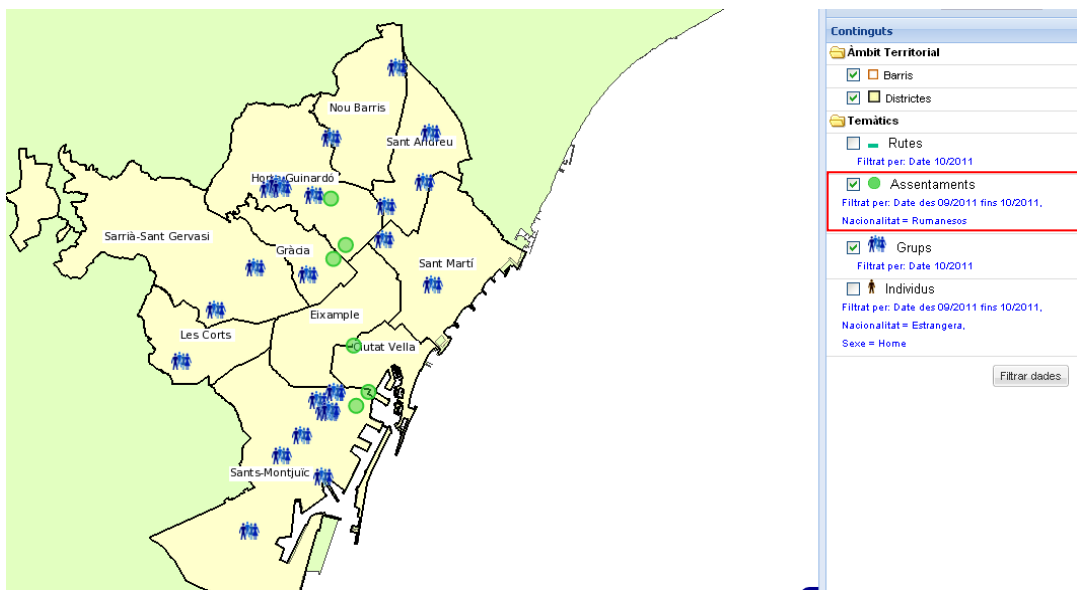


Figura 5. 18 Resultado filtro por asentamientos.

Ejemplo 3: Filtrado por rutas

Filtrar rutas de los educadores de calle de franja horaria “Tarde”, y que pertenezcan a los informes del mes de octubre del 2011. A continuación se presentan dos imágenes que permiten observar el cambio sobre la capa de rutas al aplicar el filtro.

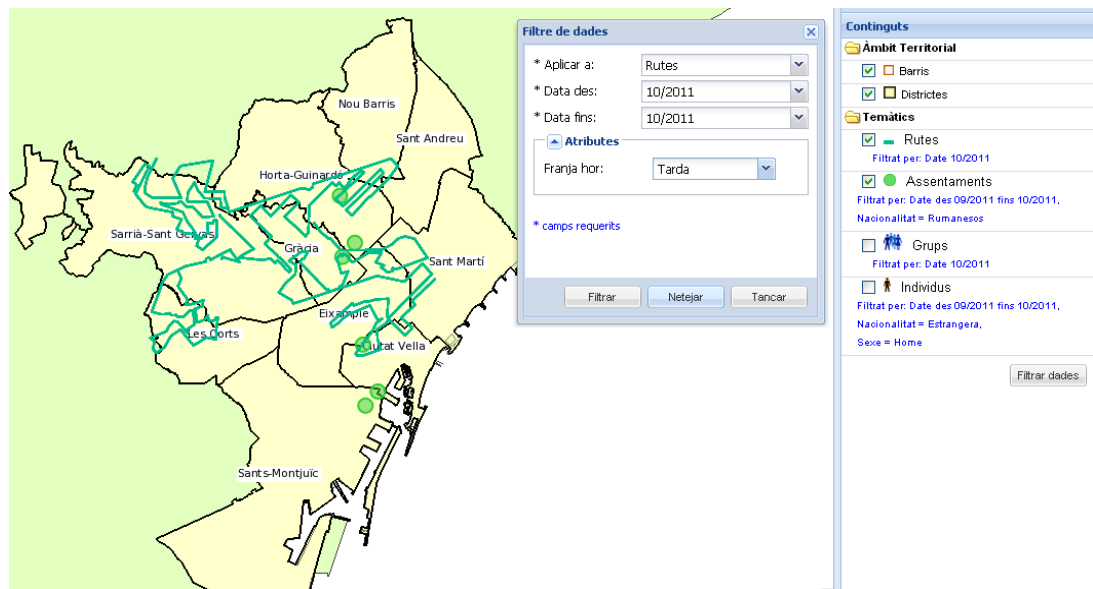


Figura 5. 19 Filtro por rutas.

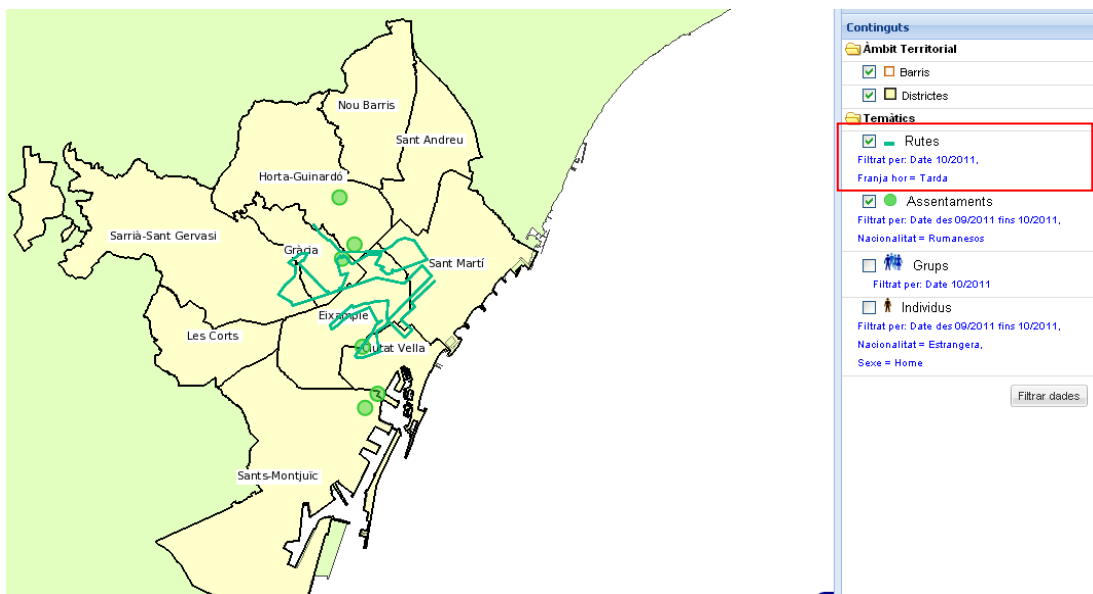


Figura 5. 20 Resultado filtro por rutas.

EXPLORACIÓN 8

Funcionalidad:

Se podrá identificar elementos de interés a través de una herramienta que mostrará los valores de los atributos relacionados.

Resultado: Para la identificación de elementos se ha incluido una herramienta novedosa que consiste en una tabla de elementos. Existe una tabla de elementos y sus atributos por cada tipo de información temática, y cada una de éstas se halla dentro de un panel de pestañas. La identificación de elementos funciona en doble sentido, se puede seleccionar un objeto

desde el mapa o desde la tabla de datos. El objeto seleccionado se pintará de color rojo para que sea rápidamente identificado por el usuario.

Nótese además que la tabla de atributos contiene solamente información de los elementos que están sobre el mapa, es decir los que corresponden al filtro realizado por el usuario. A continuación se exhiben varios ejemplos.

Ejemplo1: identificación de asentamientos.

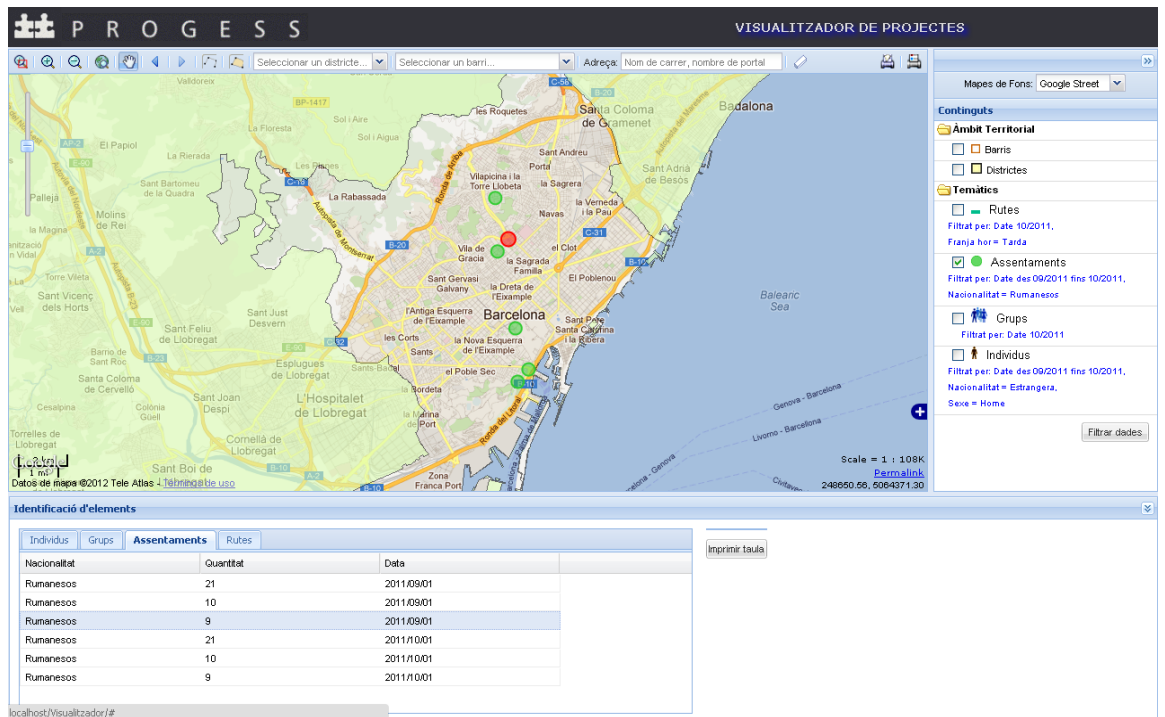


Figura 5. 21 Identificación elementos de asentamientos.

Ejemplo2: identificación de grupos.

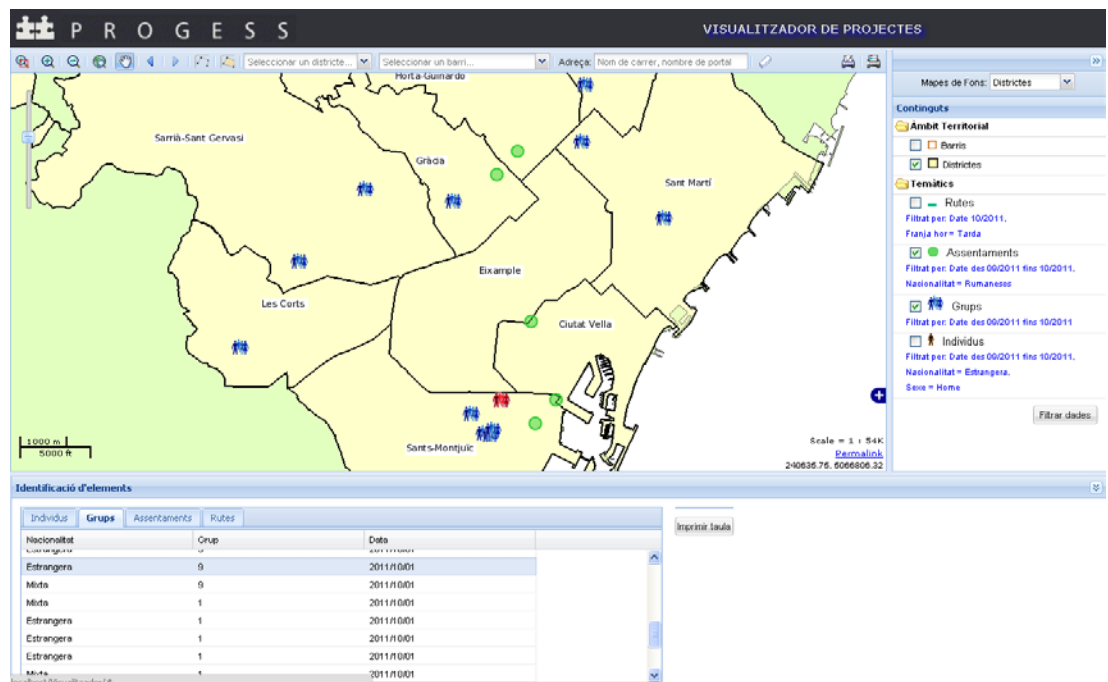


Figura 5. 22 Identificación elementos de grupos.

Ejemplo 3: identificación de individuos

En este ejemplo se muestra una selección primaria por barrio utilizando la lista de barrios de la barra de herramientas y luego una selección de elementos, utilizando la tabla de individuos del panel de identificación.

The screenshot shows the PROGRESS VISUALITZADOR DE PROJECTES interface. The map displays several districts including Sant Genís dels Agudells, El Carmel, El Coll, Vallcarca i els Penitents, La Salut, El Putxet i el Farró, Sant Gervasi - La Bonanova, and Sabo i les Planes. The 'Identificació d'elements' panel is active, showing a table of individuals.

Nacionalitat	Sexe	Data
Estrangera	D	2011/09/01
Estrangera	H	2011/09/01
Estrangera	H	2011/09/01
Estrangera	H	2011/09/01
Estrangera	H	2011/09/01
Estrangera	H	2011/09/01
Estrangera	H	2011/09/01
Estrangera	H	2011/09/01

Figura 5. 23 Identificación elementos de individuos.

Ejemplo 4: identificación de rutas

The screenshot shows the PROGRESS VISUALITZADOR DE PROJECTES interface. The map displays several districts including Nou Barris, Sant Andreu, Horta-Guinardó, Sarrià-Sant Gervasi, Gràcia, Sant Martí, Les Corts, Eixample, and Ciutat Vella. The 'Identificació d'elements' panel is active, showing a table of routes.

Franga	Data
2	2011/10/01
1	2011/10/01
1	2011/10/01
3	2011/10/01
2	2011/10/01
1	2011/10/01

Figura 5. 24 Identificación elementos de rutas.

EXPLORACIÓN 7

Funcionalidad:

Se podrá generar una ficha en formato adobe reader con los resultados de las localizaciones y filtros.

Resultado: Para llevar a cabo esta funcionalidad utilizaremos las tres herramientas de impresión con las que cuenta la aplicación: impresión de mapa, de leyenda y de tabla de elementos. A continuación se muestran ejemplos de su uso.

Ejemplo 1: Impresión de mapa



Figura 5. 25 Herramientas de impresión mapa y leyenda.

Herramienta
impresión de mapa

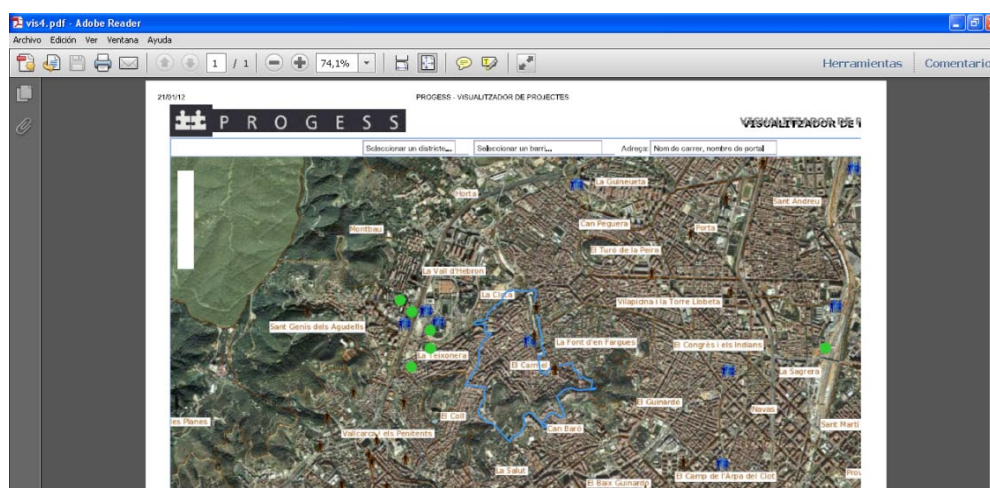
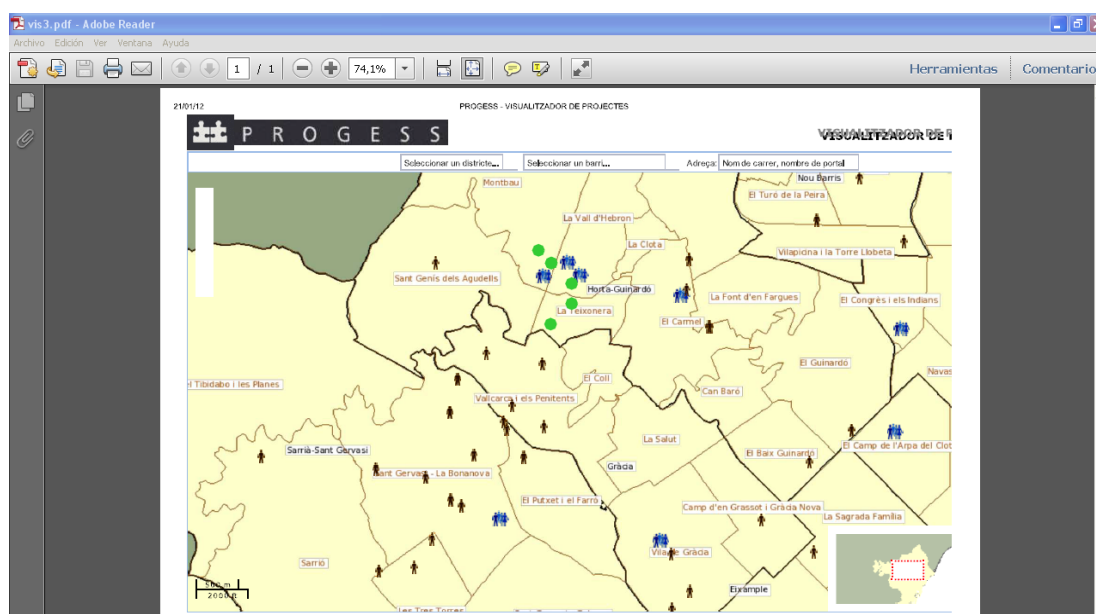


Figura 5. 26 Impresión de mapa.

Ejemplo 2: Impresión de leyenda



Figura 5. 27 Herramienta de impresión de leyenda.

Herramienta
impresión de leyenda

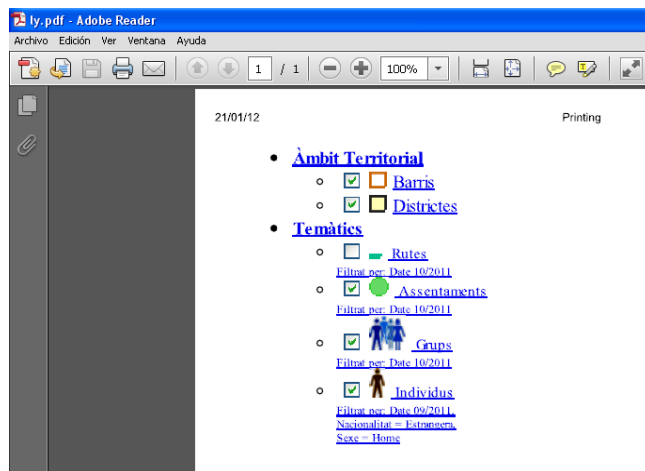


Figura 5. 28 Impresión de leyenda.

Ejemplo 3: Impresión de tabla de elementos

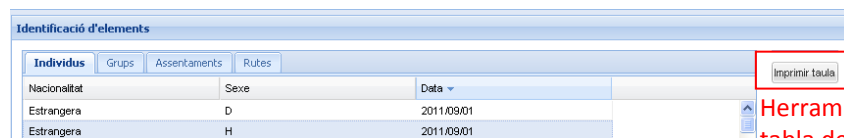


Figura 5. 29 Herramienta impresión de tabla.

Herramienta impresión
tabla de elementos

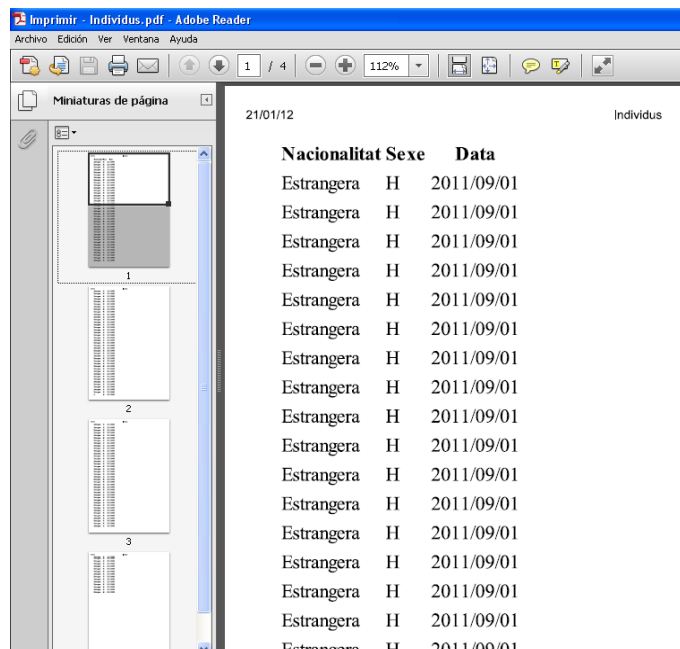


Figura 5. 30 Impresión tabla de elementos.

5.2. PRUEBAS DEL SISTEMA.

En esta sección se demuestra la consistencia de la aplicación a través de diversos tests.

PRUEBA 1

La primera prueba que realizaremos se refiere a unos de los requerimientos no funcionales del sistema: *Ofrecer compatibilidad con los navegadores de uso más común.*

A continuación se muestran los gráficos que exhiben la prueba del funcionamiento de la aplicación en los tres navegadores web más utilizados: Google Chrome, Mozilla Firefox e Internet Explorer.

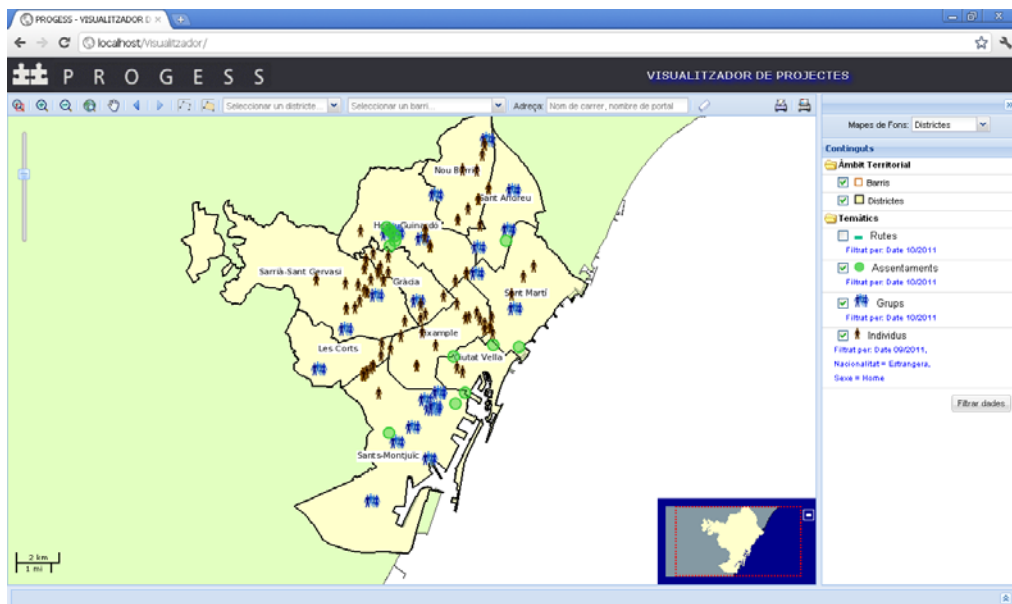


Figura 5. 31 Uso aplicación web con navegador Google Chrome.

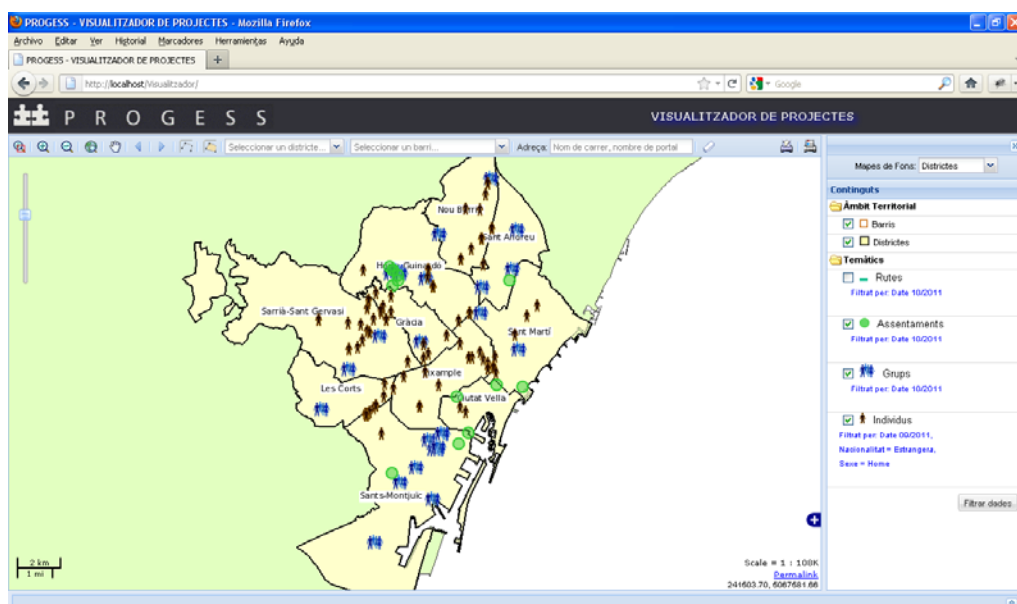


Figura 5. 32 Uso aplicación web con navegador Firefox.

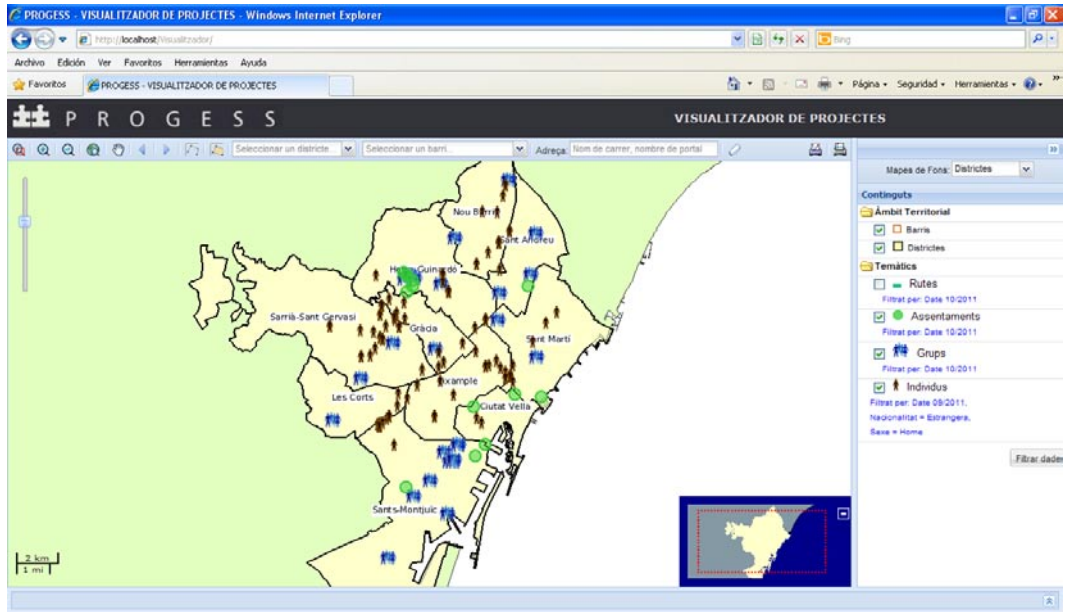


Figura 5. 33 Uso aplicación web con navegador Internet explorer.

PRUEBA 2

Esta prueba consiste en verificar que el sistema brinde al usuario mensajes de retroalimentación cuando una dirección buscada no sea encontrada dentro de los datos con los que cuenta el servidor o cuando la dirección sea ingresada incorrectamente. A continuación se muestran los resultados del test:

Campo dirección vacío:

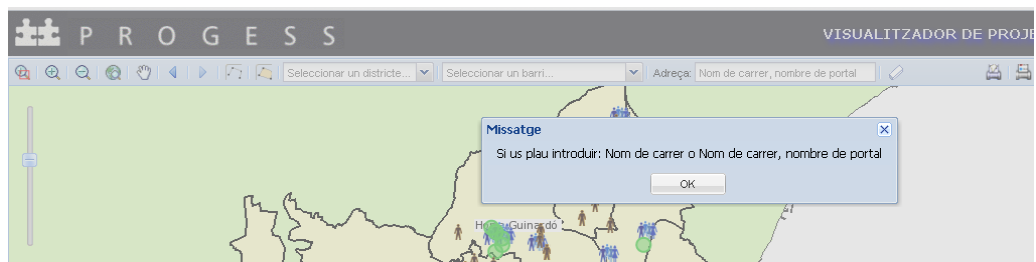


Figura 5. 34 Mensaje de error - campo dirección vacío.

Campo dirección con formato diferente al especificado (nombre de calle, número de portal):

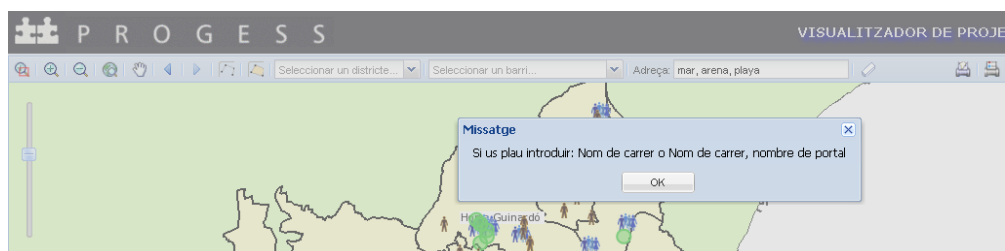


Figura 5. 35 Mensaje de error - ingreso errado dirección.

No encuentra coincidencias:

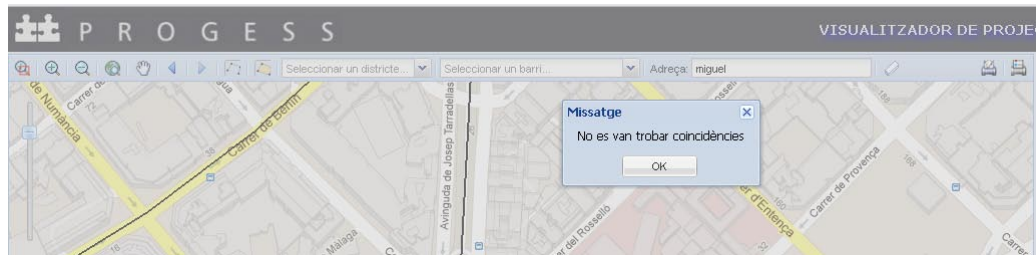


Figura 5. 36 Mensaje de error - no encuentra dirección.

PRUEBA 3

La tercera prueba consiste en verificar que el sistema brinde mensajes de retroalimentación cuando el usuario cometa errores en el formulario de filtro de datos. A continuación se muestran los resultados de estas pruebas:

Filtrar sin especificar fechas:

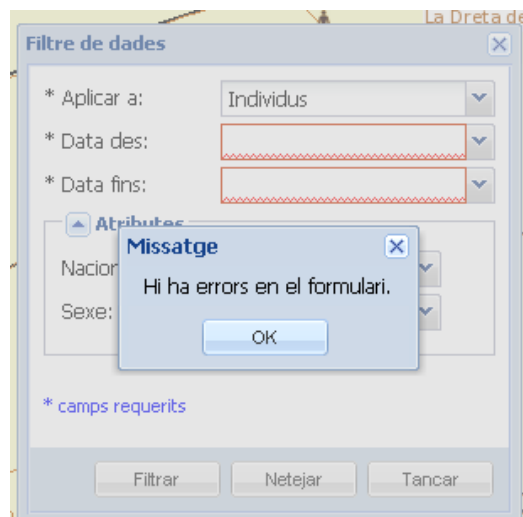


Figura 5. 37 Mensaje de error - errores en el formulario de filtro.

Filtrar sin especificar fecha desde o sin especificar fecha hasta:

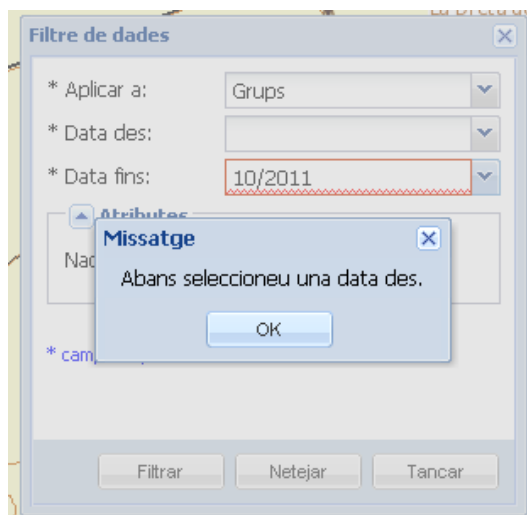


Figura 5. 38 Mensaje de error – no especifica fecha.

Filtrar especificando una fecha desde posterior a una fecha hasta:

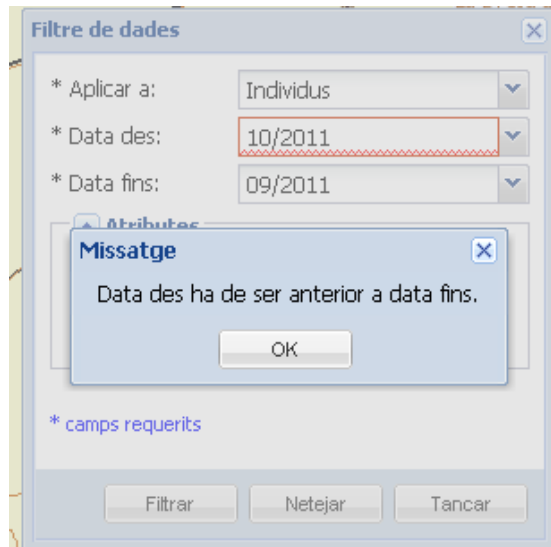


Figura 5. 39 Mensaje de error - ingreso erróneo de fecha desde.

Filtrar especificando una fecha hasta anterior a una fecha desde:

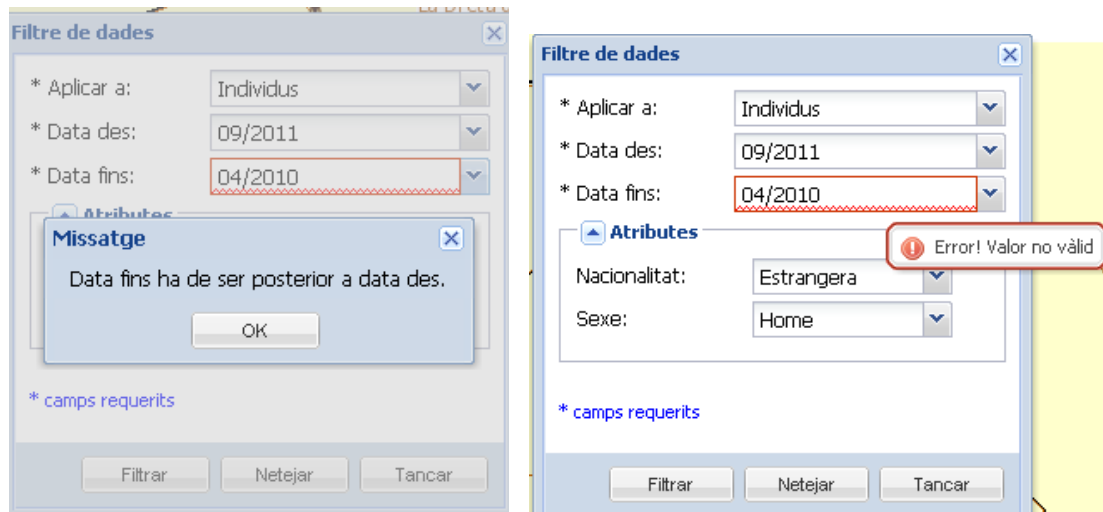


Figura 5. 40 Mensaje de error - ingreso erróneo de fecha hasta.

CONCLUSIONES

- ❖ A lo largo de este documento se han revisado y analizado cada una de las partes que conformaron la realización del Sistema de Información Geográfico tema central del proyecto final del Máster en Tecnologías de la Información Geográfica.
- ❖ Se concluye la importancia de una buena planificación y diseño del sistema. Es decir las dos primeras fases del desarrollo del sistema son cruciales para su posterior éxito.
- ❖ La utilización de las librerías scripts gratuitas, existentes en el mercado, resultan de gran apoyo para la implementación de aplicaciones clientes pues facilitan la implementación de las diferentes funcionalidades requeridas y ahorra tiempo-esfuerzo.
- ❖ El servidor de mapas MapServer ofrece una amplia gama de funciones para el tratamiento de la información espacial. La experiencia personal de la autora con referencia a éste ha sido bastante satisfactoria, sin embargo se señala la falta de una herramienta que permita facilitar las labores de simbolización dentro del servidor.
- ❖ Por último se resalta la relevancia de los conocimientos adquiridos en el máster para la consecución del presente proyecto.

BIBLIOGRAFÍA

- Gdal GeoSpatial Data Abstraction Library (2011). «GDAL main page». En: GDAL Official Web Site. Disponible en <http://www.gdal.org/index.html>.
- LONGLEY P., GOODCHILD M., MAGUIRE D., RHIND D. (2011) Geographic Information Systems & Science Third Edition John Wiley & Sons, Inc
- MapServer Open Source Web Mapping (2011). «MapServer 5.6 Documentation». En: MapServer Official Web Site. Disponible en: <http://mapserver.org/MapServer-56.pdf>.
- MySQL Server (2011). «MySQL Spatial Extensions». En: MySQL Server Official Web Site. Disponible en <http://dev.mysql.com/doc/refman/5.5/en/spatial-extensions.html>.
- Zeiler, M. (1999) Modeling Our World The ESRI® Guide to GeoDataBase Design. ESRI PRESS.

ANEXO A

GUÍA DE INSTALACIÓN

En esta sección se describe los pasos necesarios para instalar el sistema de información geográfico realizado. Incluye la instalación del servidor de mapas, servidor de base de datos, restauración de la base de datos y colocación adecuada de los archivos de la aplicación cliente y de mantenimiento. Los instaladores de las versiones utilizadas y los archivos que forman parte de las aplicaciones se encuentran en el CD que se entrega adjunto a este documento.

1. Instalar MapServer.

Para instalar el servidor de mapas utilizamos el paquete de instalación MS4W (MapServer for Windows). Se ejecuta el archivo binario y se siguen los pasos indicados, el cual permitirá instalar rápidamente y de forma sencilla un entorno de trabajo para el desarrollo con MapServer. El paquete MS4W instala un servidor pre-configurado que incluye entre otros los siguientes componentes: Servidor Web Apache, librería Gdal, soporte para PHP.

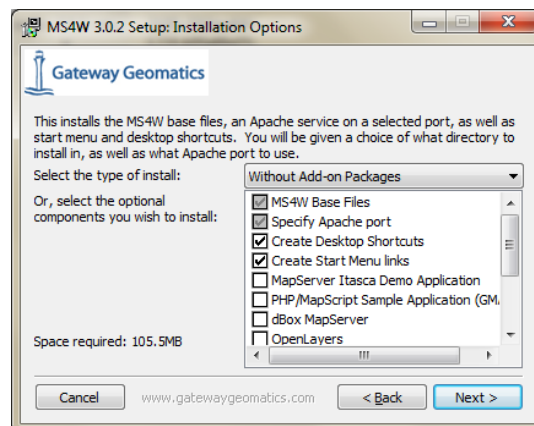


Figura A. 1 Instalación MapServer

2. Instalar MySQL y PHPMyAdmin

Como en el paso anterior es posible encontrar en la web o dentro del disco de datos que acompañan este documento los instaladores del gestor de base de datos MySQL. Una vez instalada la base de datos se recomienda también instalar la herramienta de administración para MySQL: PHPMyAdmin, para ello solo es necesario descomprimir el archivo phpMyAdmin-3.3.8.1 dentro del directorio de aplicaciones del servidor (/ms4w/apps).

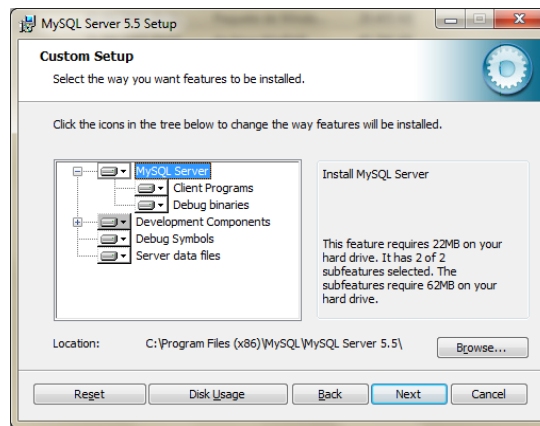


Figura A. 2 Instalación MySQL

3. En la carpeta de publicación de Apache (ms4w/Apache/htdocs) copiar los archivos phpMyAdmin.pkg, Visualizador.pkg los cuales redireccionarán las llamadas hacia los directorios de aplicaciones del servidor.
4. Se procede a restaurar la base de datos utilizando la herramienta de restauración de PHPMyAdmin y el archivo vis.sql, con el cual se creará la base de datos, las tablas, los procedimientos almacenados y se insertarán los datos iniciales necesarios.

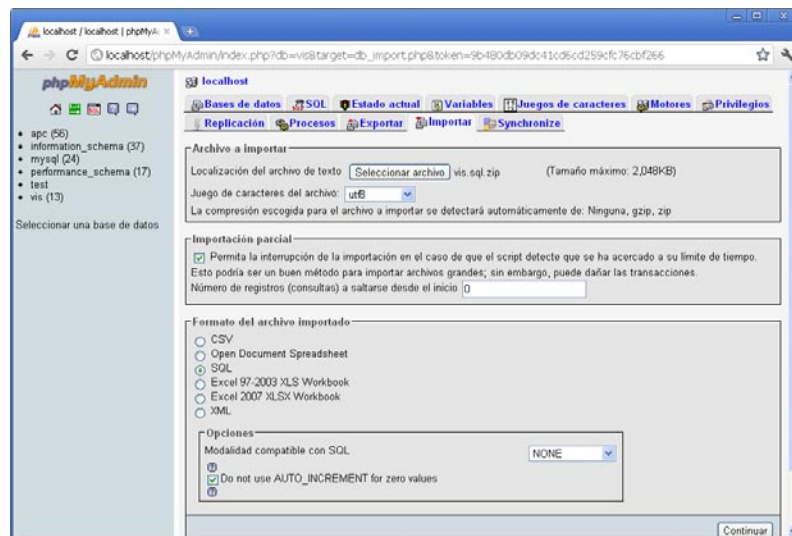


Figura A. 3 Restauracion Base de datos con PHPMyAdmin

5. Colocar los archivos de la aplicación dentro del directorio de aplicaciones web del servidor Apache. La figura anterior exhibe el sistema de archivos que conforman la aplicación web de visualización y consulta de datos espaciales.

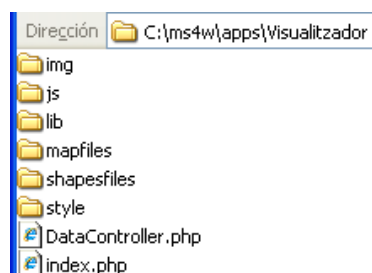


Figura A. 4 Lista de archivos de aplicación Visualizador

- La carpeta img contiene las imágenes necesarias para la interfaz de usuario y para la simbolización de las capas.
- La carpeta js contiene los archivos javascript, donde se encuentra la programación de los módulos de la aplicación.

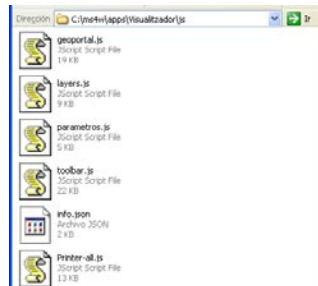


Figura A. 5 Lista de archivos JavaScript

- En la carpeta lib encontraremos las librerías utilizadas para el desarrollo del sistema, en este caso GeoExt, Ext y OpenLayers.
 - Mapfiles contiene nuestros dos archivos .map, uno para capas de divisiones administrativas (basic.map) y otro para capas temáticas (tematic.map).
 - Shapefiles contiene archivos .shp de capas bases que son llamadas por el servidor de mapas.
 - Style tiene el archivo de estilos para la interfaz web
 - En el archivo DataController.php es donde se implemente el módulo de control de datos.
 - En el archivo index.php es donde se implementa el módulo inicial y es el archivo al que invocamos cuando accedemos a la aplicación.
6. Una vez realizados los pasos anteriores comprobar que la aplicación este funcionando a través de un navegador web:
<http://localhost/Visualizador>
7. Para utilizar la **aplicación de mantenimiento** de datos solo es necesario instalar la JRE (Java Runtime Environment), programa necesario para ejecutar aplicaciones Java, y copiar el archivo ejecutable mantVis.jar que se encuentra en el CD de datos.

ANEXO B

MANUAL DE USUARIO APLICACIÓN DE MANTENIMIENTO

En este anexo se detalla el modo de utilización del programa de actualización, el cual está conectado a la base de datos del sistema en MySQL.

Para ejecutar la aplicación utilizaremos el archivo ejecutable mantVis.jar, el cual desplegará una ventana con dos tablas de datos. La primera nos muestra la lista de información temática territorial existente y la segunda exhibe la lista de informes del tipo de información seleccionada de la primera tabla, como es posible ver en la siguiente figura.

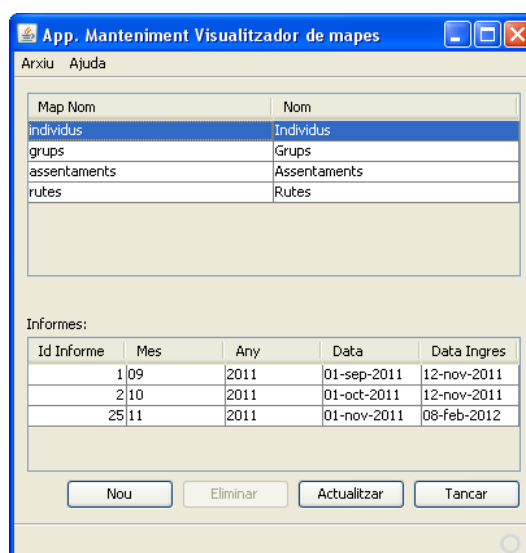


Figura B. 1 Ventana principal aplicación de mantenimiento.

El programa permite realizar dos funcionalidades para el mantenimiento de los datos: crear un nuevo informe y eliminar un informe.

Creación de Informes:

1. Para agregar un nuevo informe a la base de datos se utiliza la opción Nou disponible desde el menú Arxiu o desde el botón de la ventana principal.
2. La opción Nou desplegará una nueva ventana en la cual definiremos los datos necesarios para ingresar un nuevo informe. En este formulario es necesario seleccionar un tipo de información temática, el mes al que pertenecen los datos y el fichero de datos espaciales y alfanuméricos que deberá estar en formato shape. La aplicación ofrece ayuda al usuario para realizar esta operación a través de mensaje indicativos de error o éxito en la

ejecución de la operación. A continuación se exhiben algunas imágenes de ejemplos de la ejecución.

Ejemplos retroalimentación de errores:

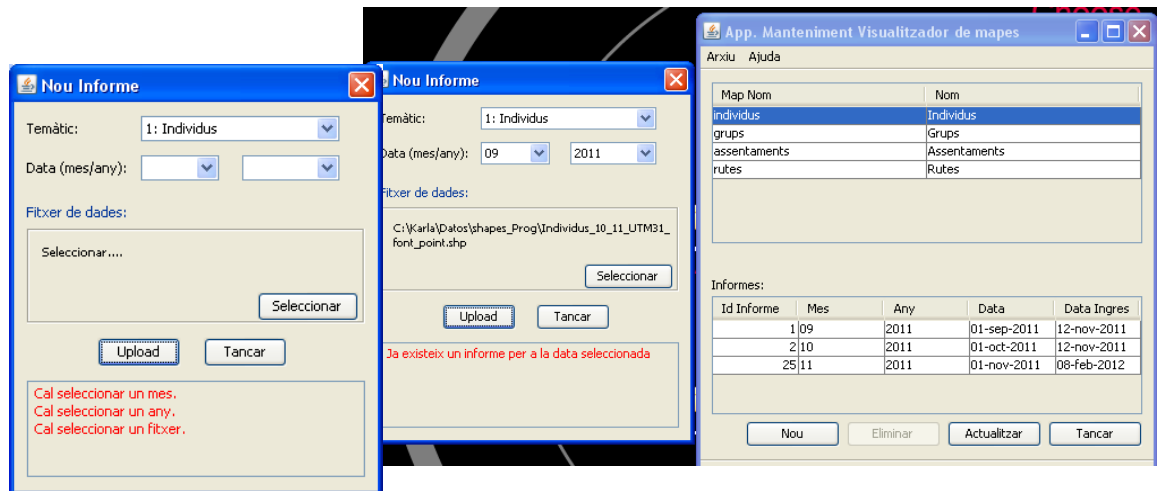


Figura B. 2 Retroalimentación de errores al crear un informe.

Ejemplos de éxito en la operación:

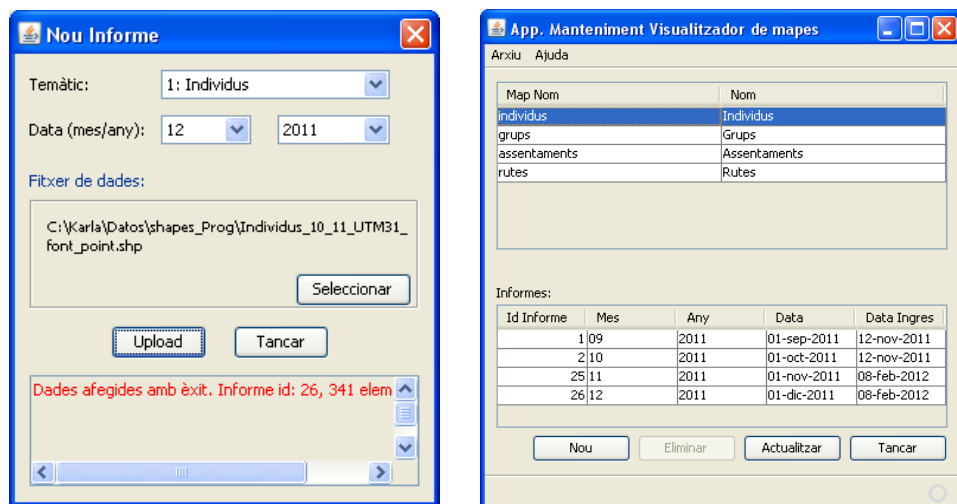


Figura B. 3 Operación de creación de informe exitosa.

Una vez realizada la operación también podemos verificar que el ingreso de datos haya sido éxito revisándolo en nuestra aplicación web:

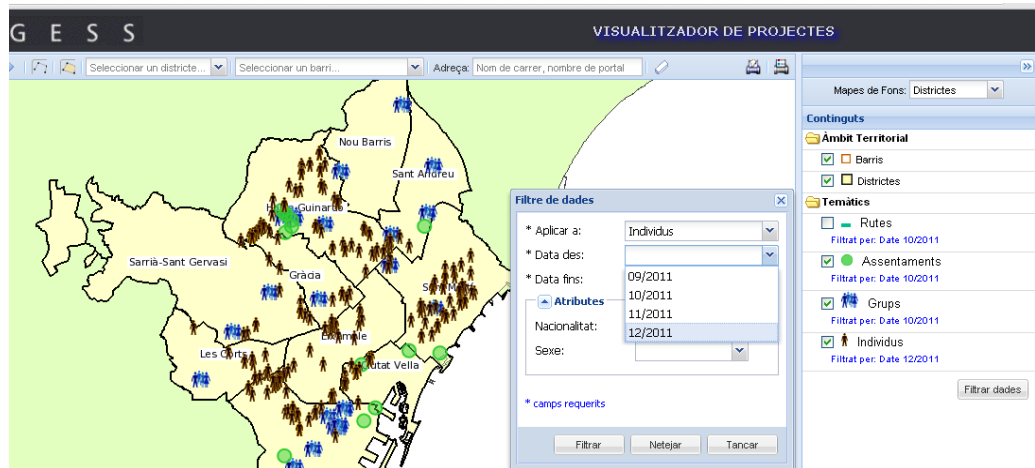


Figura B. 4 Verificación de ingreso de nuevo informe.

Eliminación de Informes:

1. Para eliminar un informe existente en la base de datos se utiliza la opción Eliminar disponible desde el menú Arxiu o desde el botón de la ventana principal.
2. La opción Eliminar mostrará un mensaje de confirmación antes de ejecutar la acción. Si el usuario confirma se realizará la eliminación y la tabla de informes ya no mostrará el registro eliminado e inexistente. Las siguientes imágenes muestran la ejecución y verificación de la operación:

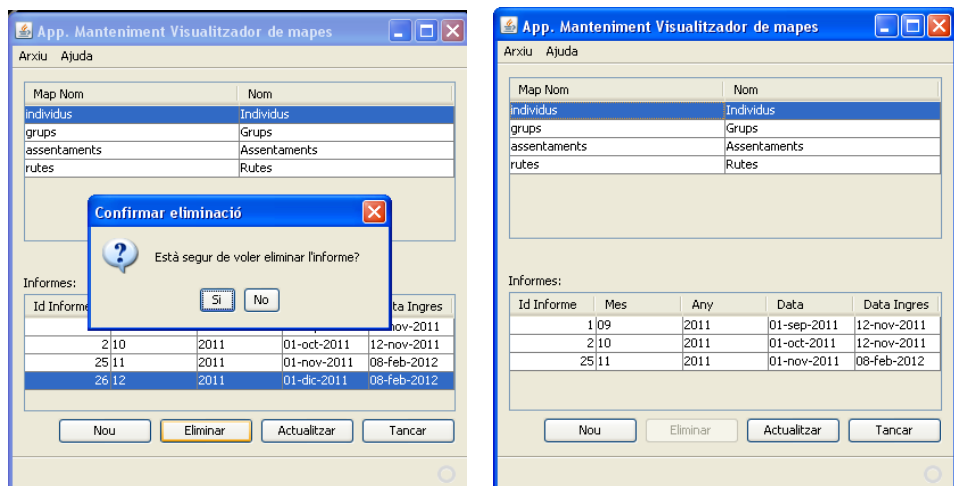


Figura B. 5 Eliminación de informe.

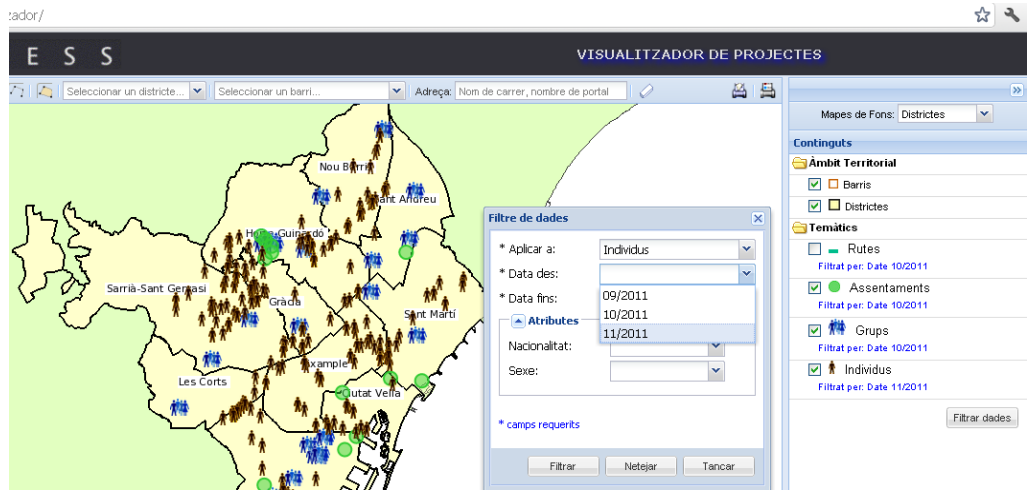


Figura B. 6 Verificación de eliminación de informe.